MAX PLANCK GRADUATE CENTER
MIT DER
JOHANNES GUTENBERG-UNIVERSITÄT

DISSERTATION

Multiscale Modeling and Deep Learning: Reverse-mapping of Condensed-phase Molecular Structures

Dissertation zur Erlangung des Grades
„Doktor rerum naturalium (Dr. rer. nat.)“

in den Fachbereichen

08 - Physik, Mathematik und Informatik
09 - Chemie, Pharmazie, Geographie und Geowissenschaften
10 - Biologie und der
Universitätsmedizin
der
Johannes Gutenberg-Universität

Marc STIEFFENHOFER
3. Juni 2022

Erstgutachter:
Prof. Dr. Kurt Kremer
Max-Planck-Institut für Polymerforschung

Zweitgutachter:
Prof. Dr. Michael Wand
Johannes Gutenberg-Universität Mainz

Betreuer:
Prof. Dr. Kurt Kremer
Dr. Tristan Bereau
Prof. Dr. Michael Wand

Datum der mündlichen Prüfung:
04.07.2022

Declaration of Authorship

I, Marc STIEFFENHOFER, declare that this thesis titled, “Multiscale Modeling and Deep Learning: Reverse-mapping of Condensed-phase Molecular Structures” and the work presented in it are my own. I hereby declare that I wrote the dissertation submitted without any unauthorized external assistance and used only sources acknowledged in the work. All textual passages which are appropriated verbatim or paraphrased from published and unpublished texts as well as all information obtained from oral sources are duly indicated and listed in accordance with bibliographical rules. In carrying out this research, I complied with the rules of standard scientific practice as formulated in the statutes of Johannes Gutenberg-University Mainz to insure standard scientific practice.

Signed:

Date:

"... the sciences do not try to explain, they hardly even try to interpret, they mainly make models. By a model is meant a mathematical construct which, with the addition of certain verbal interpretations, describes observed phenomena. The justification of such a mathematical construct is solely and precisely that it is expected to work—that is, correctly to describe phenomena from a reasonably wide area."

John von Neumann

Abstract

Molecular processes can be studied at various levels of resolution that range from the fundamental, quantum mechanical description of electronic degrees of freedom up to the classical thermodynamic description of macroscopic quantities. For many systems, and in particular for those incorporating macromolecules, a single model is not able to capture all the relevant length- and timescales to thoroughly study a phenomena of interest. Multiscale modeling (MM) offers a solution by combining molecular models at different resolutions to address phenomena at multiple scales. On the low-resolution end, coarse-grained (CG) models are deployed to study the large-scale behavior of the system. These CG models are constructed by averaging over atomistic degrees of freedom. Their low resolution reduces the computational effort of the simulation and enables a faster exploration of configuration space. In addition to coarse-graining, a tight and consistent link between models of different resolutions calls for a reverse-mapping capable of reintroducing degrees of freedom as well. Reverse-mapping is routinely applied in the MM community, for example to compare simulation results with experimental data, to rigorously analyze the simulation results on a local scale, or to assess the stability and accuracy of the obtained CG structures. At the heart of this work is the development of deepbackmap (DBM), an approach for the reverse-mapping of condensed-phase molecular structures. The new method is based on machine learning (ML), a study of computer algorithms that use data to construct statistical models. Traditional schemes start from a rough coarse-to-fine mapping, which requires further energy minimization and subsequent molecular dynamics simulations to equilibrate the system. DBM directly predicts equilibrated molecular configurations that agree with the Boltzmann distribution. Moreover, DBM requires little human intervention, as the reintroduction of details is learned from training examples. During the course of this thesis, DBM is applied to various tasks involving reverse-mapping: The general performance and transferability of DBM is evaluated at the example of a polymeric system consisting of polystyrene molecules. Beside an excellent accuracy of structural properties for reverse-mapped configurations, DBM displays a remarkable transferability across different state points and chemical space. Moreover, reverse-mapping with DBM is performed to assess the quality of CG models at the atomistic resolution. In addition, DBM is applied to adjust local structural properties, such as bond lengths and angles, of configurations obtained with top-down molecular models in order to resemble target distributions obtained with structure-based models more closely. Finally, a ML-based scheme inspired by DBM is applied for temporal coherent reverse-mapping of molecular trajectories. Overall, this thesis demonstrates the advantages of integrating generative ML methods into the framework of MM, especially for problems that are difficult to solve from a pure physics-based perspective.

Zusammenfassung

Molekulare Prozesse können auf unterschiedlichen Auflösungsstufen untersucht werden, die von der quantenmechanischen Beschreibung elektronischer Zustände bis hin zu der klassischen Beschreibung makroskopischer Eigenschaften reichen. In einigen Fällen, insbesondere wenn Makromoleküle untersucht werden, ist ein einzelnes Modell jedoch nicht ausreichend, um alle relevanten Längen- und Zeitskalen eines Phänomens zu erfassen. Multiskalen Modellierung (MM) bietet hierfür eine Lösung, indem mehrere molekulare Modelle mit unterschiedlicher Auflösung kombiniert werden. Coarse-grained (CG) Modelle mit einer geringen Auflösung werden genutzt, um das Verhalten des Systems auf großen Skalen zu erfassen. Die geringere Auflösung reduziert den notwendigen Rechenaufwand der Simulation und ermöglicht eine schnellere Untersuchung des Konfigurationsraumes. Zusätzlich sind umgekehrte Abbildung, die es erlauben Freiheitsgrade zurück zu gewinnen, ebenfalls wichtig im Bereich von MM. Eine Erhöhung der Auflösung ist beispielsweise häufig notwendig, um einen direkten Vergleich von Simulationsdaten mit experimentellen Ergebnissen anzustellen oder um einen Startpunkt für weitere hochaufgelöste Simulationen zu erhalten. Im Zentrum dieser Doktorarbeit steht die Entwicklung von Deepbackmap (DBM), eine Methode für die Erhöhung der Auflösung von molekularen Systemen in der kondensierten Phase. Die Methode stützt sich auf maschinelles Lernen (ML), eine Wissenschaft von Computeralgorithmen, die statistische Modelle von Daten ableiten. Traditionelle Ansätze starten von ungenauen Anfangskonfigurationen, die Energie Minimierung und anschliessende Equilibrierung erfordern. DBM hingegeben ermöglicht es, direkt equilibrierte molekulare Strukturen zu erzeugen, die sich im Einklang mit der Boltzmannverteilung befinden. Des Weiteren benötigt DBM nur wenig menschliches Eingreifen, da die Zurückgewinnung von Freiheitsgraden anhand von Beispielen gelernt wird. Zunächst wird DBM an einem polymerischen System aus Polystyren getestet. Es wird demonstriert, dass DBM nicht nur molekulare Strukturen von hoher Qualität generieren kann, sondern auch, dass DBM eine erstaunliche Generalisierbarkeit bezüglich unterschiedlicher Phasen und chemischer Systeme aufzuweisen hat. Anschliessend wird gezeigt, dass DBM für eine Qualitätsbewertung von CG Modellen auf atomistischer Auflösung benutzt werden kann. Des Weiteren wird DBM angewendet, um lokale strukturelle Eigenschaften von molekularen Strukturen zu adjustieren, um eine gegebene Verteilung eines top-down Modelles näher an eine Zielverteilung eines strukturbasierten CG Modelles anzugleichen. Zum Schluss wird eine von DBM inspirierte Methode eingeführt, die eine zeitlich koherente Erhöhung der Auflösung von molekularen Trajektorien ermöglicht. Zusammengefasst demonstriert diese Arbeit, wie generatives ML im Bereich von MM erfolgreich eingesetzt werden kann.

Acknowledgements

I would like to take this opportunity to express my gratitude to several people who accompanied me at this journey. First of all, I want to thank Prof. Bernhard Mehlig from the Chalmers university in Gothenburg. Attending his course on artificial neural networks during a semester abroad was the starting point for my interest in the research field of artificial intelligence. At the same time, I want to highlight the impact that Prof. Friederike Schmid and PD. Peter Virnau had on my scientific carrier due to their great masters course *statistical mechanics and computer simulations* at the JGU Mainz. This course awakened my interest in statistical physics and molecular simulations. Ultimately, this course lead me to my masters thesis in the KOMET 1 research group, where I got the opportunity to merge my interests for machine learning and statistical mechanics under the careful supervision of Prof. Friederike Schmid and Prof. Michael Wand. In fact, Prof. Michael Wand became my mentor for everything machine learning related from this point on until now, as he continued his great supervision during the course of my PhD. Thank you for joyful discussions, deep insights and illuminating explanations during all this time!

I am also extremely thankful to Prof. Kurt Kremer and Dr. Tristan Bereau for welcoming me into the theory group at the MPIP for my PhD. Prof. Kurt Kremer has established a productive and yet warm and friendly atmosphere in his research group that allowed me to further push my limits and develop as a scientist. I have to express my sincere gratitude to Dr. Tristan Bereau, who became my supervisor at the MPIP. His limitless energy, expertise and motivation was constantly inspiring me and enabled me to accomplish my scientific goals. Even after he quit university for a new adventure, he still found time to guide me through the final phase of my PhD. Thank you for your great supervision!

I also want to thank former and current members of the theory research group at the MPIP. At first, I want to mention Dr. Clemens Rauer and Dr. Kiran Kanekal, who became my office mates and helped me with the first steps of my PhD. Furthermore, I want to thank Dr. Christoph Scherer, Dr. Alessia Centi, Dr. Chan Liu, Dr. Roberto Menichetti, Dr. Yasemin Bozkurt Varolgunes, Bernadette Mohr, Dr. Arghya Dutta, Dr. Joseph Rudzinski and Dr. Martin Girard for both, great scientific discussions during various seminars as well as casual discussions during lunch. All of you made the time at the MPIP a pleasure.

A special thanks goes to my collaborators Moritz Hoffmann, Kirill Shmilovich, Nick Charron, Dr. Christoph Scherer, Daniel Franzen, Dr. Falk May and Dr. Denis Andrienko. I also want to thank Dr. Joseph Rudzinski, Dr. Martin Girard, Dr. Denis Andrienko, Dr. Lukas Kades, Dr. Joydip Chaudhuri and Dr. Christoph Scherer for proof-reading chapters of this thesis.

Finally, I want to acknowledge the Max Planck Graduate Center in Mainz as my funding source. In addition, this work was supported by the TRR 146 Collaborative Research Center of the Deutsche Forschungsgemeinschaft. I also want to thank the

Institute for Pure and Applied Mathematics (IPAM) at the UCLA for inviting me to the long program *Machine Learning for Physics and the Physics of Learning*, where I got the opportunity to spend time in a great environment and to have inspiring discussions with scientists from all over the world.

Last but not least, I want to include some personal acknowledgments here. I would have never gotten to this point without the continues support of my loving partner and family. Thank you for never letting me down and for encouraging me to follow my passion.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The exploration of molecular processes is fundamental to a wide range of modern research areas, such as polymer science [1], drug design [2] or folding dynamics of proteins [3]. Novel algorithms and high-performance computing have made computational chemistry an important tool to gain further insights into the molecular nature of matter [4]. In particular, the significance of physics-based models has to be highlighted, which are purposeful simplifications of molecular systems that are too complex to be solved analytically. Molecular modeling facilitates calculations and predictions about the structure, thermodynamics and dynamics of molecular systems [5].

On the other hand, processing of complex and high-dimensional data has become a hallmark of modern machine learning (ML). In the past decades, ML has emerged as a prominent research field that has a transformative impact on many domains, such as computer vision [6], speech recognition [7] or medical image analysis [8]. At its core, ML algorithms construct statistical models from data without relying on explicit program instructions. As such, the recent success of ML models is further fueled by the availability of large data sets. Recently, ML is gaining significant attention in many fields of modern science as well, especially particle physics and computational chemistry [9, 10, 11].

This thesis explores the advantages of integrating ML methods into molecular simulation frameworks, especially for problems that are difficult to solve from a pure physics-based perspective. ML is already applied in the computational chemistry community frequently, for example to construct molecular potentials [12] or for the analysis of simulation data [13]. Here, the emphasis is on generative tasks, i.e. deploying ML models to learn the complex dependencies between particles in order to synthesize realistic molecular structures. In particular, generative ML algorithms originally designed for computer vision are applied to increase the resolution of coarse-grained molecular systems. However, before the concept and goal of this thesis are outlined in more detail, the scientific context of this work has to be established first.

The theoretical underpinning for molecular models is given by statistical mechanics, which successfully explains macroscopic properties of matter in terms of microscopic degrees of freedom. However, while the fundamental principles to describe the motions of microscopic particles are well known, i.e. quantum-mechanics or Newton's equation of motion, the enormous number of microscopic degrees of freedom makes analytical solutions for most molecular systems intractable. In addition, resolving the microscopic state of a molecular system experimentally displays resolution limits: Modern microscopy techniques, such as cryo-EM [14, 15] or X-ray crystallography [16], achieve a spatial resolution of a few angstrom. However, a thorough understanding of molecular processes, such as protein folding, additionally requires a high temporal resolution [17]. Microscopy techniques with high temporal resolution, like PALM [18] or LLSM [19], typically yield a lower spatial resolution. While longer exposure times or shorter wavelengths could be used to increase the spatiotemporal resolution in the diffraction experiments, the induced radiation damage prevents applications to biological systems [20].

A possible remedy is offered by computer models based on experimental observations and/or analytical approximations. Simulations of the model can be used to study the behavior of the system and to predict its properties. The resolution limit of such computer models is theoretically only bound by computational effort. The most fundamental description of matter is the quantum-mechanical description that includes electronic degrees of freedom. However, models at this level of detail are computationally very demanding. As an example, a popular method is density functional theory deploying B3LYP functionals [21, 22], which scales as $\mathcal{O}(N^3)$, where N is the number of atoms in the model system [23]. The computational cost can be reduced significantly, when the molecular resolution is reduced to the level of single atoms that are treated as hard spheres. In this approximation, the effect of electrons is modeled as a potential energy surface representing the quantum ground-state. Such models are routinely implemented by molecular dynamics (MD) simulations that numerically integrate Newton's equation of motion. The deployed atomic interactions are often empirical and aim at correctly modeling structural, thermodynamic and/or dynamic properties of a target system [24]. The computational effort of such classical models is dominated by long-range interactions, such as van der Waals and electrostatic interactions. Typically particle mesh Ewald summation is used to reduce the computational cost to $\mathcal{O}\big(N\log(N)\big)$ [25].

Rapid fluctuations of the atoms typically require an integration time step in the range of femtoseconds [26]. However, timescales of relevant biological processes, such as protein folding or binding, can be in the order of microseconds up to seconds [27, 28]. Therefore, an extremely large number of integration steps is required, such that even dedicated hardware and specialized software reach their limits: Current state-of-the-art integration systems achieve hundreds of nanoseconds up to tens of microseconds of simulation data per day for molecular systems that contain a few thousands of atoms [29, 30].

To push the limits of accessible length- and timescales in the computer simulation, a coarser description of matter is routinely used. To this end, coarse-grained

(CG) variables are deployed that represent an average over atomistic degrees of freedom. The lower resolution of CG systems reduces the computational effort of the simulation and enables larger integration time steps [31, 32]. In addition, dynamics of the CG system are typically accelerated due to "softer" interactions between CG sites [33, 34]. As such, CG models enable a faster exploration of configuration space.

The particular resolution of a molecular model depends on the length- and timescales of the phenomena of interest. However, some phenomena display a wide range of relevant scales and therefore can not be captured by a single model. This is especially true for soft matter systems, such as polymers, where processes on multiple scales can be linked and interwoven [35, 36, 37]. In particular, local interactions can impact large-scale conformational changes. Consequently, molecular modeling of soft matter systems requires a methodology that can capture the interplay of processes that are potentially linked to various different scales.

A solution is offered by multiscale modeling (MM), where models of different resolutions are combined to address phenomena at multiple scales [38, 39, 36]. At the lower end, CG molecular models are deployed to study the large-scale behavior of the system. However, a tight and consistent link between models of different resolutions requires to accomplish both mapping directions. In particular, a reverse-mapping to reintroduce details is required for the following reasons: (1) To rigorously analyze the simulation results on a local scale [40, 41, 42, 43], (2) to enable a direct comparison to experimental data, for example obtained with spectroscopic methods [44], (3) to serve as starting point for further high-resolution simulations [45, 41], or (4) to assess the stability and accuracy of the obtained CG structures [45].

The fine-to-coarse mapping of molecular configurations is typically a straightforward computation, such as the center-of-mass calculation for a set of atoms. However, reverse-mapping is more challenging, as new degrees of freedom have to be generated taking all their dependencies into account. In particular, a reverse-mapping scheme has to fulfill the following requirements: (1) The reintroduced details have to be consistent with the CG conformation, i.e. coarse-graining of the reverse-mapped structure has to yield the original CG structure. (2) The generated microstates must have high statistical weight and should ideally follow the Boltzmann distribution. (3) In addition, the mapping should not be unique, as the reduced resolution implies that a single CG structure corresponds to an ensemble of atomistic microstates.

Reverse-mapping is widely used in the molecular modeling community and several approaches to reintroduce details exist. Most reverse-mapping schemes follow the same strategy: At first, an initial atomistic structure is generated that is consistent with the given CG structure. Two major approaches exist for this step: (1) Generic approaches place atoms close to their corresponding CG site, either randomly or based on geometric rules [46, 47]. (2) Fragment-based schemes rely on a presampled library of atomistic fragments that are projected onto the CG conformation [37, 48, 44, 49]. In both cases, energy minimization to relax the initial structure is required. Subsequently, MD simulations are performed to recover the correct statistical weights of the reinserted degrees of freedom.

The computational effort for the subsequent energy minimization and equilibration procedures of such reverse-mapping schemes can become significant. As such, applications to large systems or high-throughput simulations are still limited. In addition, poorly initialized structures can get trapped into local minima with high energy barriers. Therefore, human intervention is frequently required for the reverse-mapping of more complex molecular structures and hence, hinder the automation of such processes.

ML has shown its ability to detect and reproduce complex dependencies in a wide range of different domains. In particular, deep neural networks (DNNs) have received considerable attention in the field of computer vision [6]. For example, generative models based on DNNs are able to synthesize photorealistic images of complex objects, such as human faces or animals [50, 51, 52]. At its core, the great success of DNNs can be linked to a multiscale approach: Multiple layers are arranged subsequently and each layer transforms its input into a more abstract and composite representation. As such, DNNs represent data with multiple levels of abstraction.

Recently, generative DNNs have been deployed in a conditional framework [53]. In particular, labels or cartoons of objects have been used as a conditional variable for the model in order to generate a corresponding high-resolution image. Generating a high-resolution image from a low-resolution representation has striking similarity to the reverse-mapping task of molecular structures. Is it therefore possible to take advantage of DNNs for MM?

This thesis answers this question with a resounding yes and demonstrates ML-based reverse-mapping. However, many challenges have to be solved to successfully accomplish this task. For example, how to define a training objective for the reverse-mapping? How can molecular structures be represented? How to avoid memory issues for large, high-dimensional configurations? All of these questions will be addressed in this work ultimately leading to the development of deepbackmap (DBM), a DNN-based approach for the reverse-mapping of condensed-phase molecular structures. In order to fulfill the consistency criteria, the CG variables are used as a conditional input for the ML model. Unlike other backmapping schemes, DBM aims at directly predicting equilibrated molecular structures resembling the Boltzmann distribution. Therefore, no further energy minimization or MD simulations are required. In addition, DBM requires little human intervention, since the reinsertion of local details is learned from training data.

A flowchart for the structure of this thesis can be found in Fig. 1.1. The first two main chapters establish the theoretical foundation for this work. In particular, chapter 2 reviews the multiscale modeling approach, including statistical mechanics, molecular dynamics, coarse-graining and the challenges of reverse-mapping. Chapter 3 gives an introduction to ML with an emphasis on DNNs and generative

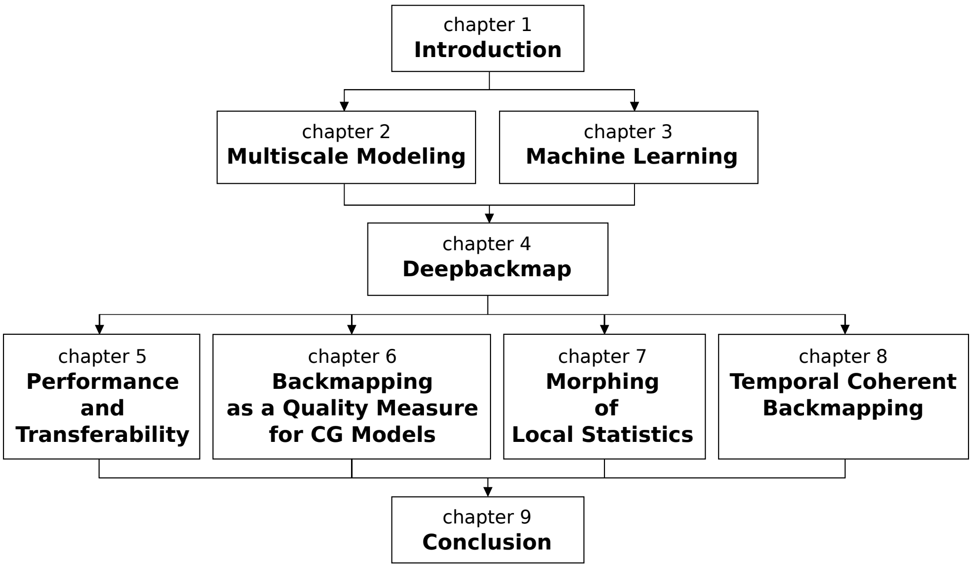

FIGURE 1.1: Flowchart of this thesis' chapters. The theoretical foundation of this work is given by chapter 2 and 3. The core of this thesis is formed by chapter 4, where the methodology of deepbackmap is introduced. Applications of DBM and its variants can be found in chapters 5-8.

models. The core of this thesis forms chapter 4, where the methodology of DBM is introduced and important concepts for the ML-based reverse-mapping task are outlined. In subsequent chapters, DBM and other ML-based techniques are applied to multiscale simulations: The general performance and transferability of DBM is evaluated in chapter 5 at the example of a challenging condensed-phase polymeric system that consists of polystyrene molecules. Moreover, chapter 6 deploys reverse-mapping with DBM to assess the quality of CG models at the atomistic resolution. In chapter 7, DBM is applied to adjust local structural properties, such as bond length and angles, of structures obtained with top-down molecular models in order to resemble a target distribution obtained with structure-based models more closely. In chapter 8, a ML-based scheme inspired by DBM is applied to the reverse-mapping of molecular trajectories aiming at temporal coherence between subsequent frames. Finally, the thesis is concluded in chapter 9, where the highlights of this work are reviewed and future research questions are posed.

Chapter 2

Multiscale Modeling

Phenomena of condensed matter can be studied at various levels of resolution that range from the fundamental, quantum mechanical description of electronic degrees of freedom up to the classical thermodynamic description of macroscopic quantities. Ideally, all emergent phenomena of matter should be treated by ab initio methods, i.e. methods based on first principles. However, even performed on modern supercomputers, ab initio molecular dynamics simulations quickly reach their limits and are currently restricted to systems involving a few thousands of atoms [54, 55]. Therefore, it is often necessary to deploy a coarser description of matter in order to push the limits of accessible length- and timsescales.

The choice of resolution depends on the length- and timsescales of the phenomena of interest. Ideally, the applied model is able to capture all length- and timsescales that are relevant for the emergent phenomena. However, in some cases the relevant scales are too far apart from each other and can not be captured in a single model. This is especially true for soft matter systems, where processes linked to atomistic length- and timsescales can lead to mesoscopic or even macroscopic changes. Whether a spontaneous change of the system is favorable or forbidden is indicated by the sign of the change in free energy $F = U - TS$, where U is the internal energy, T the temperature and S the entropy. The rather low characteristic energy scale of soft matter systems is in the order of magnitude of the thermal energy, $k_b T$[35, 36, 37]. Therefore, entropic contributions to the free energy due to large scale conformational and structural changes can be in the same order of magnitude as local interactions. Thus, soft matter systems are characterized by large thermal fluctuations. Consequently, a thorough exploration of soft matter systems demands for methods that capture the interplay of processes that are potentially linked to various different scales.

A solution is offered by *Multiscale Modeling* (MM), which is illustrated in Fig. 2.1. MM is a method that combines models at different resolutions in order to address phenomena at different length- and timsescales [38, 39, 36]. At the lower end, a coarse-grained (CG) model is deployed that eliminates degrees of freedom, while aiming at reproducing specific features of a target system, such as structural or thermodynamic properties. The reduced representation decreases molecular friction, smooths the energy landscape, and thereby effectively accelerates sampling of the

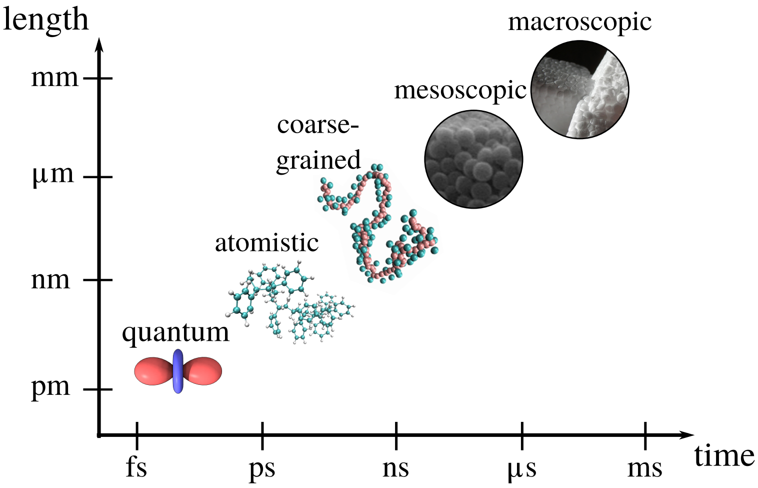

FIGURE 2.1: Multiscale Modeling of soft matter at the example of polystyrene. Various levels of resolution are shown: From a quantum mechanical description of the electronic structure up to a macroscopic scale. Illustration at the mesoscale is taken from [56].

conformational space. However, the MM approach also includes the other direction and deploys strategies to switch back to a higher resolution model when required. In order to establish a tight and consistent link between models at different resolutions, various strategies can be used: (1) In the sequential approach models are treated separately and information is passed between them without directly influencing each other [57, 58], whereas (2) hybrid methods provide a direct interaction between models allowing to use different resolutions simultaneously [59, 60, 61]. (3) Alternatively, the resolution of single molecules can be changed adaptively during the course of the simulation [62, 63]. In this thesis, the focus is set to sequential MM.

This chapter is an introduction to MM and is organized as follows: At first, basics of thermodynamics and statistical mechanics are recalled followed by a review of molecular dynamics simulations. Afterwards, a section about coarse-graining outlines strategies to reduce the resolution in order to extend accessible length- and timsescales. Finally, the inverse problem, i.e. increasing the resolution, is introduced to motivate the main theme of this thesis.

2.1 Thermodynamics and Statistical Mechanics

Classical thermodynamics describes the behavior of bulk, macroscopic systems in terms of a few macroscopic quantities, such as the total internal energy E, the total volume V, and the number of particles N. Typically, the system is considered at *thermodynamic equilibrium*, where average properties become time-invariant. In particular, the actual state of a system at thermodynamic equilibrium is history-independent, i.e. properties of the system only depend on the current conditions of state and not on its preparation.

The basic concepts of classical thermodynamics were developed before the molecular nature of matter was generally accepted. In fact, the laws of classical thermodynamics are based only on a few postulates without referencing to a more fundamental description on the molecular level [64]. As such, it is not surprising that classical thermodynamics is concerned with laws and relationships exclusively for macroscopic quantities.

The molecular underpinning of thermodynamics was developed in the field of *statistical mechanics*, where all the microscopic details of individual molecules are taken into account. For example, in the classical picture, a list of positions $\mathbf{r} \in \mathrm{R}^{3N}$ and momenta $\mathbf{p} \in \mathrm{R}^{3N}$ of N atoms are considered, whereas a quantum mechanical description uses quantum states. In the following, the classical picture is used for simplicity and a microscopic state $m = (\mathbf{r}, \mathbf{p})$ is characterized by its $6N$ degrees of freedom, i.e. a point in the $6N$ dimensional *phase space*.

The following introduction to thermodynamics and statistical mechanics is largely based on the textbook [64] by Shell.

2.1.1 The Mircrocanonical Ensemble

In statistical mechanics, macroscopic quantities measured at equilibrium are described as the average behavior of many particles. For an isolated system, i.e. a system that can not exchange energy or particles with its surrounding at fixed volume, a macrostate is completely specified by (E, V, N), which remains constant throughout molecular motion [64]. For each macrostate (E, V, N), a collection of possible microstates can be found, i.e. a surface in the phase space of N atoms with constant total energy E at a volume V. This collection of microstates together with their associated probabilities is called the *microcanonical ensemble*.

The positions and velocities of the atoms constantly vary under the influence of their mutual interactions. Therefore, the microstate changes constantly even if the macrostate stays fixed. The likelihood that a microstate will be visited by the system is denoted with p_m. Note that the microstate probabilities do not change with time at equilibrium. A cornerstone of statistical mechanics is the statement that the system has no preference for a certain microstate and hence, each microstate is equally likely [64]. This fundamental rule is called the *principle of equal a priori probabilities*. It allows to write the likelihood in the canonical ensemble as

$$p_m = \begin{cases} \frac{1}{\Omega(E,V,N)} & \text{if } E_m \neq E \\ 0 & \text{if } E_m \neq E \end{cases}, \tag{2.1}$$

where $\Omega(E, V, N)$, called the *density of states*, is a function describing the number of accessible microstates for a particular macrostate (E, N, V) [64].

2.1.2 Entropy and the Second Law of Thermodynamics

A central theme common for thermodynamics, statistical mechanics as well as information theory is the concept of *entropy*. In classical thermodynamics, the entropy S is regarded as a non-conserved state-function that emerges naturally for systems in equilibrium [64]. It is a function

$$S = S(E, V, N) \tag{2.2}$$

dependent on the macroscopic quantities E, V and N. Allowing for heat, volume or mass transfer, the system can change its equilibrium macrostate to another macrostate. This is called a *thermodynamic process*. Historically, entropy was introduced to explain why some thermodynamic processes are irreversible, i.e. the process occurs spontaneously in one direction, whereas the reverse does not, although both directions obey the conversation of energy [65]. The reason for this is the tendency of thermodynamic systems to progress towards states with increasing entropy. This is stated in the second law of thermodynamics: The entropy of an isolated system can not decrease as it always evolves to an equilibrium state where the entropy is highest [64].

While the specific form of the entropy function is different for every system, all entropy functions have some shared properties. One of the most important is the total differential

$$dS = \frac{1}{T}dE + \frac{P}{T}dV - \frac{\mu}{T}dN, \tag{2.3}$$

which relates the temperature T, the pressure P and the chemical potential μ to derivatives of the same function. As such, T,P and μ are not independent in the entropy function and can be derived from (E, V, N) [64].

The above definition for the entropy S is exclusively based on macroscopic properties. Boltzmann was the first who gave a definition for the entropy based on microscopic considerations and therefore introduced a connection of thermodynamics to the molecular nature of matter [64]. His famous formula reads

$$S = k_B \ln(\Omega(E, V, N)), \tag{2.4}$$

where k_B is a proportionality constant, called *Boltzmann's constant*. Eq. 2.4 links the entropy S to the number of accessible microstates for a given macrostate. Therefore, the second law of thermodynamics can be interpreted as the tendency of a system to evolve to a state that maximizes the number of accessible microstates.

Based on Boltzmann's equation, Gibbs introduced a more general form of the entropy

$$S = -k_B \sum_m p_m \ln(p_m). \tag{2.5}$$

Note that upon application of the principle of equal a priori probabilities, i.e. $p_m = \frac{1}{\Omega(E,V,N)}$, the entropy is maximized and Gibbs formulation of the entropy recovers

Boltzmann's equation.

2.1.3 The Canonical Ensemble

Central to the previous considerations was the ensemble for a system in isolation, i.e. the microcanonical ensemble for fixed (E, V, N). In the following, the *canonical ensemble* is introduced, which describes a system that is not isolated but at constant temperature. To this end, the system is considered to be in thermal contact with an infinitely large heat bath with a fixed temperature T. Therefore, the fixed macroscopic quantities of the system are (T, V, N), while the total energy E is allowed to fluctuate. The composite of the system and the heat bath is again considered to be an isolated system. Summing over the microstates of the heat bath allows to derive the probabilities for the microstates m of the system of interest. Importantly, the microstate probabilities are no longer equal but depend on their total energy E_m. More specifically, the microstate probabilities can be written as

$$p_m = \frac{\mathrm{e}^{-\frac{E_m}{k_B T}}}{Z}, \tag{2.6}$$

where the normalization constant

$$Z = \sum_m \mathrm{e}^{-\frac{E_m}{k_B T}} \tag{2.7}$$

is called the *canonical partition function* and the probability distribution in Eq. 2.6 is referred to as *Boltzmann distribution* [64]. Similarly to the microcanonical distribution, the Boltzmann distribution is the distribution that maximizes the entropy for a given macroscopic state (T, V, N). In general, the canonical ensemble is used more frequently as the microcanonical ensemble, since in most cases systems are considered that are in thermal equilibrium with their surroundings.

2.1.4 Thermodynamic Limit and Statistical Equivalence of Ensembles

The canonical approach provides an alternative, in addition to the microcanonical approach, to determine the behavior of a system at a microscopic level. While there are rigorously no fluctuations in the energy in the microcanonical ensemble, energy fluctuates in the canonical ensemble but the temperature is rigorously constant. However, in the *thermodynamic limit*, i.e. when the number of particles and the volume of the system go to infinity $N \to \infty$, $V \to \infty$ while the particle density is held fixed $\frac{N}{V} =$ constant, the differences in macroscopic properties for both ensembles vanish [64].

This can be seen clearly when the distribution of the energy in the canonical ensemble is considered. Using Eq. 2.6 and 2.4 the probability for a specific energy E

can be written as

$$p(E) \propto \Omega(E,V,N)e^{-\frac{E_m}{k_BT}} = e^{\frac{1}{k_B}(S(E,V,N)-\frac{E_m}{T})}. \tag{2.8}$$

This equation indicates that two competing terms have to be taken into account for the probability of a specific energy level: The first term is the entropy S, which is a concave, increasing function of E [64]. The second term $-\frac{E}{T}$ decreases linearly with the energy E. Therefore, the probability distribution for the energy levels has a maximum at an intermediate energy E^*. Both terms, S and E, are extensive quantities, i.e. they scale as N. Since the competing terms are within the exponential, the probability distribution becomes sharply peaked at the maximum $p(E^*)$. Therefore, $p(E^*)$ becomes the most dominant term and the impact of microstates with different energies $E \neq E^*$ vanishes [64].

Moreover, fluctuations of the total energy E become extremely small in the thermodynamic limit. The variance of the total energy $\sigma_E^2 = \langle E^2 \rangle - \langle E \rangle^2$ can be linked to the heat capacity, which is an extensive quantity. Therefore, the relative magnitude of energy fluctuations scales as

$$\frac{\sqrt{\sigma_E^2}}{E} \propto N^{-1/2}. \tag{2.9}$$

As a consequence, a macroscopic system appears to have constant total energy [64].

2.1.5 Information-theoretic View on Statistical Mechanics

Information theory is the mathematical study of the coding, transmission, storage and quantification of information [66]. A central concept in information theory is the quantification of the amount of uncertainty for the outcome of a random process. In 1948, Shannon introduced the information entropy S_{Shannon} for a random variable X

$$S_{\text{Shannon}} = -\sum_{i=1}^{n} x_i \ln(x_i), \tag{2.10}$$

where $x_1, \dots, x_n$ are possible outcomes of X [67]. In particular, Shannon has shown that S_{Shannon} is a quantity that increases with increasing uncertainty [67]. In 1957, Jaynes published two papers emphasizing the correspondence between information theory and statistical mechanics [68, 69]. Jaynes stated that Gibbs' entropy (Eq. 2.5) in statistical mechanics and Shannon's information entropy were identical except for the Boltzmann constant [68]. Moreover, statistical mechanics could be viewed from the perspective of information theory, such that deriving microscopic distributions could be treated as an inference problem [68].

Jaynes treated the testable information, i.e. the macroscopic observables, as prior information [68]. However, information is missing required to determine the specific microstate of a system due to a description of the system solely in terms of macroscopic quantities, i.e. a description that is too coarse. Consequently, when probabilities are assigned to microstates two requirements have to be fulfilled: (1) The microscopic probability distribution has to be consistent with the observed macroscopic quantities, i.e. the correct average behavior has to be captured [68]. (2) The ignorance of the specific microstate the system resides has to be taken into account without introducing arbitrary assumptions [68]. If only macroscopic properties of the system are known, a microscopic distribution has to be assumed that is the least-informative in order to express the uncertainty of the actual microstate. From this point of view, the principle of equal a priori probabilities is a consequence of the ignorance about the microscopic details of the system.

Jaynes states that the microscopic distribution can be found by variational principles [68]: A microscopic distribution has to be found that maximizes the information entropy under the constraints of the observed macroscopic properties. In practice, the problem is solved using Lagrange multiplier to impose the constraints. For the microcanonical ensemble, a constraint has to be applied that assigns zero probability to microstates with total energies that differ from the observed total energy [68]. In the canonical ensemble, the constraint is a fixed expectation value for the energy [68]. In both cases, the well known results from statistical mechanics are reproduced, i.e. the uniform distribution for the microcanonical ensemble and the Boltzmann distribution for the canonical ensemble.

2.2 Molecular Dynamics Simulation

Physicists aim at expressing the basic laws of nature in terms of relatively simple equations. However, solving such equations analytically, like the equations of motion for more than two interacting bodies, becomes intractable in most cases. This is unfortunate, as it is central to the statistical mechanics view on thermodynamics to consider systems with an extremely large number of interacting particles. Computer simulations can circumvent these issues by numerically predicting the behavior of a model system. Two main branches of computer simulation techniques for molecular systems are molecular dynamics (MD) and Monte Carlo (MC). While MD simulations mimic molecular movements by numerically integrating Newton's equation of motion for the molecules, MC simulations are based on random sampling and statistical probabilities of acceptance/rejection of moves. In the following, the focus is set on MD simulations.

MD is a simulation technique to evolve the system for a fixed period of time in order to sample the conformational space representatively. In particular, the time evolution of the system is discretized by small time steps δt. After the system is initialized with prespecified positions $\mathbf{r}$ and momenta $\mathbf{p}$, it is propagated forward in time. To this end, the microstate is updated at every step based on the forces $\mathbf{f} = \frac{d\mathbf{p}}{dt}$ acting on the particles.

MD is widely used to study equilibrium and dynamic properties of a system. To this end, the system is brought to equilibrium, i.e. it is evolved for a sufficient amount of time, such that macroscopic properties do not change anymore. Afterwards, snapshots or whole trajectories of the system can be extracted and the actual measurement of the observable quantity of interest can be performed. In this regard, MD simulation builds a bridge between theory and experiment, as it enables to test a model and compare it with experimental results [70].

The following introduction to MD is based on the book [71] by Frenkel and Smit and the lecture notes [72] of Shell, which are excellent sources providing the interested reader with more detailed explanations on this topic.

2.2.1 Numerical Integration

Consider a molecular system that consists of N particles with positions $\mathbf{r} \in \mathbb{R}^{3N}$, momenta $\mathbf{p} \in \mathbb{R}^{3N}$ and masses $\mathbf{m} \in \mathbb{R}^{N}$. The Hamiltonian H of the system is the function that gives the total energy of a microstate,

$$H(\mathbf{p}, \mathbf{r}) = K(\mathbf{p}) + U(\mathbf{r}), \tag{2.11}$$

where $K(\mathbf{p}) = \sum_i \frac{|\mathbf{p}_i|^2}{2m_i}$ is the kinetic energy and $U(\mathbf{r})$ the potential energy. The time evolution of the system is described by Newton's equation of motion

$$\frac{d\mathbf{p}_i}{dt} = -\frac{dU(\mathbf{r})}{d\mathbf{r}_i}. \tag{2.12}$$

Eq. 2.12 is a set of $3N$ second-order, nonlinear, coupled partial differential equations, which is intractable to be solved analytically [72]. Therefore, numerical integration is used to evolve the system in time. The basic idea is to introduce a small time step δt and update the positions of the atoms at consecutive time steps, i.e.

$$\mathbf{r}(0), \mathbf{r}(\delta t), \mathbf{r}(2\delta t), .. \tag{2.13}$$

Many different algorithms exist to propagate the system forward in time. As an example, the *Verlet* algorithm is explained in the following.

Using a Taylor expansion, the position at time $t + \delta t$ can be written as

$$\mathbf{r}(t+\delta t) = \mathbf{r}(t) + \frac{d\mathbf{r}(t)}{dt}\delta t + \frac{d^2\mathbf{r}(t)}{dt^2}\frac{\delta t^2}{2} + \frac{d^3\mathbf{r}(t)}{dt^3}\frac{\delta t^3}{6} + \mathcal{O}(\delta t^4). \tag{2.14}$$

Similarly, the position at the previous time step can be written as

$$\mathbf{r}(t-\delta t) = \mathbf{r}(t) - \frac{d\mathbf{r}(t)}{dt}\delta t + \frac{d^2\mathbf{r}(t)}{dt^2}\frac{\delta t^2}{2} - \frac{d^3\mathbf{r}(t)}{dt^3}\frac{\delta t^3}{6} + \mathcal{O}(\delta t^4). \tag{2.15}$$

Adding Eq. 2.14 and 2.15 and rearranging leads to

$$\mathbf{r}(t+\delta t) = 2\mathbf{r}(t) - \mathbf{r}(t-\delta t) + \frac{d^2\mathbf{r}(t)}{dt^2}\frac{\delta t^2}{2} + \mathcal{O}(\delta t^4). \tag{2.16}$$

Eq. 2.16 enables to compute the positions at the next time step from the positions at the two previous time steps and the forces, which can be calculated using Eq. 2.12. Starting from initial positions and velocities, the time-evolution of the system traces a path in phase space, called trajectory. For classical systems, the phase space trajectory lies on a surface of constant energy, as Newton's equations conserve energy [72]. For ergodic systems, the trajectory will eventually visit all points in phase space that are in agreement with the given total energy, whereas systems that are non-ergodic have areas in phase space that are inaccessible. While it is not possible to sample all states of the trajectory, MD simulations aim at sampling the accessible phase space representatively. In particular, the averages of MD simulations for ergodic systems can represent the thermodynamic properties at the macroscopic scale.

Note that the Verlet algorithm has two important features: It is time-reversible and symplectic, i.e. volume in phase space is preserved [70]. Those features are crucial to maintain correct statistical sampling and stability, as those are properties of the true Hamiltonian dynamics [72]. Algorithms that do not preserve the volume in phase space can dramatically expand the initial volume, such that it eventually covers areas of phase space that are not compatible with the starting condition and might violate energy conservation. Although reversibility and the conservation of phase space volume does not automatically guarantee that there is no drift of the total energy on a long timescale, it is at least a reasonable requirement [71]. In practice, the Verlet algorithm does not exactly conserve the total energy, but it exhibits only small long energy drifts [71]. Note, that there are many different algorithms that can be derived from the Verlet algorithm yielding identical trajectories, such as the leap frog or the velocity Verlet algorithm [71].

2.2.2 Molecular Force Fields

The physical accuracy of a MD simulation relies largely on the method by which the forces are specified, i.e. on the potential energy function $U(\mathbf{r})$ that defines the interaction between the particles. Typically, $U(\mathbf{r})$ is referred to as molecular force field. Depending on the representation of the system, $U(\mathbf{r})$ can be defined on various different levels of resolution. At the most fundamental level, a quantum description of the system is deployed that includes electronic degrees of freedom [73, 74]. In the following, the classical description is used, which ignores the motion of the electrons and focuses solely on the motion of the nuclei. More specifically, the classical description approximates the effect of electrons as a potential energy surface representing the quantum ground-state [72]. This description is based on two major approximations: (1) The nuclei are treated as point particles that follow classical Newtonian dynamics. This is reasonable because they are much heavier than electrons [72]. (2) The Born-Oppenheim approximation is applied, which states that electrons and nuclei can be treated separately, because the dynamics of the electrons are so fast that they can be considered to react instantaneously to the motion of their nuclei [72].

Most of the force fields used in classic MD are empirical and aim at correctly modeling structural, thermodynamic and/or dynamic properties of the system [24]. The potential $U(\mathbf{r}; \mathbf{P})$ usually consists of simple analytic functions with a set of parameters $\mathbf{P}$, which are specified by fitting $U(\mathbf{r}; \mathbf{P})$ to experimental data or detailed electronic calculations [75, 76, 77]. Typically, the potential is divided into two terms

$$U = U_{\text{bonded}} + U_{\text{nonbonded}}, \tag{2.17}$$

one term representing bonded interactions, i.e. interactions associated with covalent bonds, and one term representing non-bonded interactions [72]. This distinction arises due to different interpretations of the solution of the Schrödinger equation, which is the fundamental differential equation that describes the wave function of a quantum-mechanical system.

Bonded Interactions

When atoms approach at close range they can share pairs of electrons and form a stable electron configuration. As a result, a covalent bond is formed, i.e. a balance of attractive and repulsive forces occurs between both atoms that binds them to each other. In order to mimic covalent interactions in a simulation, the following potentials are typically applied: (1) Bond stretching accounts for deviations from the equilibrium distance between two bonded atoms, (2) bond angle bending accounts for deviations from the preferred hybridization geometry and (3) bond torsion/dihedral accounts for rotations along bonds. The high energy scales associated with bond stretching and bending only allows for small deviations from the equilibrium bond length and bond angle. As such, Taylor expansions around the minimum can be applied yielding harmonic potentials. Note that such harmonic potentials do not account for bond breaking/forming [72]. Torsional interactions are typically associated with energy scales lower than bond stretching or bending and are approximated by a cosine expansion. In summary, a frequently used analytical form to model bonded interactions is

$$U_{\text{bonded}} = \sum_{\text{bonds}} a(d - d_0)^2 + \sum_{\text{angles}} b(\Phi - \Phi_0)^2 + \sum_{\text{dihedrals}} \left(\sum_n c_n \cos(\omega)^n \right), \tag{2.18}$$

where d is the bond length, Φ the bond angle and ω the dihedral angle [72]. Similarly, d_0 is the equilibrium bond length and Φ_0 the equilibrium bond angle. The parameters a, b and c_n are the strength for the harmonic and the cosine series potential, respectively.

Non-bonded Interactions

The non-bonded potential is associated with van der Waals attraction, Pauli repulsion and electrostatic interactions [72]. The van der Waals attraction arises due to correlations between instantaneous dipoles of the electron clouds of the atoms. The Pauli repulsion stems from overlapping electron clouds and is a consequence of the Pauli principle, which forbids any two electrons from having the same quantum numbers. Both, the van der Waals attraction and the Pauli repulsion, are often combined into a single expression, such as the Lennard-Jones potential. The electrostatic forces arise due to partial or formal charges of the atoms and are taken into account through Coulomb's law. In combination, a typical non-bonded potential is modeled as

$$U_{\text{nonbonded}} = \sum_{\text{pairs}} \Bigg(\underbrace{4\epsilon\left[\left(\frac{r_{ij}}{\sigma}\right)^{-12} - \left(\frac{r_{ij}}{\sigma}\right)^{-6}\right]}_{\text{Lennard-Jones}} + \underbrace{\frac{q_i q_j}{4\pi\epsilon_0 r_{ij}}}_{\text{Coulomb}} \Bigg), \tag{2.19}$$

where r_{ij} is the pairwise distance, q_i and q_j are the net (partial) charges, ϵ_0 is the electric permittivity in vacuum, σ is the distance at which the Lennard-Jones potential reaches its minimum value $-\epsilon$ [72]. Note that the Lennard-Jones parameters σ and ϵ depend on the particular atom types [72].

Eq. 2.19 is an effective pair potential that approximates many-body interactions by renormalizing the pairwise interactions in order to limit the computational expense. However, calculating non-bonded interactions is computationally more expensive compared to bonded interactions, since the number of terms in the pairwise atomic sum scales as N^2, while the other scale as N. To reduce the computational overhead, i.e. to avoid quadratic scaling of the non-bonded interactions, the Lennard-Jones potential is often truncated as its contribution becomes minimal for large distances. To this end, a cutoff distance of $r_c \approx 2.5\sigma$ is typically used where the energy is only a few percent of the minimum energy ($U_{\text{LJ}}(r_c) \approx -0.016\epsilon$) [72]. In addition, it is common practice to shift the potential to avoid discontinuities, i.e. subtracting the value of the potential at the cutoff. However, the truncated contributions can become significant for the total energy and pressure of the system. To this end, a correction to the total potential can be introduced, which is derived analytically for isotropic systems [72, 71]. On the other hand, the long-range Coulomb interaction requires a special treatment, as a tail correction can not be derived directly [72]. To this end, particle mesh Ewald summation is typically used to reduce the computational effort to the order of $N\log(N)$[25]. In this case, the potential is split into short-range and long-range contributions. Short-range contributions are computed in real space, while long-range contributions are computed in Fourier space.

2.2.3 Controlling Temperature and Pressure

The numerical integration scheme described in Sec. 2.2.1 enables sampling from the microcanonical ensemble, i.e. maintain a constant total energy. However, it is often desirable to sample from different ensembles, such as the canonical ensemble (NVT) or the isothermal-isobaric ensemble (NPT). Modifications on the MD algorithm to control the temperature or pressure are called thermostat or barostat algorithms, respectively. They are used to match experimental conditions, study temperature dependent processes or enhance the efficiency of conformational search [78]. Several methods exist to control temperature and pressure during the simulation. In the following, some popular techniques are introduced.

Thermostats

The simplest approach to control the temperature is the velocity rescaling algorithm. The temperature is related to the kinetic energy and can be estimated as

$$T = \frac{2\langle K \rangle}{k_{\mathrm{B}} n_{\mathrm{DOF}}}, \tag{2.20}$$

where K is the kinetic energy and n_{DOF} are the degrees of freedom [72]. Therefore, the velocities can be rescaled at each time step to fix the temperature to a desired value. Despite its simplicity, this algorithm does not reproduce the correct thermodynamic properties of the canonical ensemble, which allows the kinetic energy to fluctuate. However, deploying velocity rescaling at every step will fix the kinetic energy, i.e. fluctuations in the kinetic energy are not captured [72].

Anderson introduced random collisions of the molecules with an imaginary heat bath at the desired temperature. To this end, particles are chosen at random and their velocities $\mathbf{v}$ are sampled randomly from the Maxwell-Boltzmann distribution

$$p(\mathbf{v}) = \Big(\frac{m}{2\pi k_B T}\Big)^{3/2} \exp\Big[-\frac{m|\mathbf{v}|^2}{2k_B T}\Big]. \tag{2.21}$$

Although this approach generates the correct canonical ensemble probabilities, the molecular kinetics are not reproduced correctly, because the random collisions decorrelate the system [72].

Nosé augmented the Hamiltonian with two extra degrees of freedom representing an imaginary heat bath:

$$H_{\text{Nosé}} = \sum_i^N \Big(\frac{\mathbf{p}_i^2}{2m_i s^2}\Big) + U(\mathbf{r}) + \frac{p_s^2}{2Q} + (3N+1)\frac{\ln(s)}{k_b T} \tag{2.22}$$

Here, s is the position and p_s is the momentum of the heat bath [71]. The parameter Q is an effective mass associated with s, i.e. $p_s = Q\frac{\mathrm{d}s}{\mathrm{d}t}$ and its magnitude determines the coupling between the heat bath and the original system. It has to be chosen carefully by the user, as it influences the temperature fluctuations [78]. Using the Lagrangian, it can be shown that the particles are coupled to the heat bath by scaling the momenta:

$$\mathbf{p}_i = m_i \mathbf{v}_i \cdot s \tag{2.23}$$

The Hamiltonian in Eq. 2.22 can then be used to derive the equations of motions for the extended system, i.e. for both, the heat bath and the original system. Note that this approach is deterministic, as no stochastic element is present.

While the Nosé thermostat generates the correct thermodynamics for the canonical ensemble, scaling of the particle momenta using the position of the imaginary heat bath also implies scaling of the timescale in the extended system [72]. Since the position s is variable, the implied timescale also changes making it difficult to implement the Nosé thermostat. To solve this issue, Hoover proposed an alternative by replacing the heat bath momentum p_s with a friction coefficient $\xi = \frac{\mathrm{d}ln(s)}{\mathrm{d}t}$:

$$H_{\text{Nosé-Hoover}} = \sum_i^N \left(\frac{\mathbf{p}_i^2}{2m_i s^2} \right) + U(\mathbf{r}) + \frac{\xi^2 Q}{2} + 3Nk_B T \ln(s) \tag{2.24}$$

This approach is known as the Nosé-Hoover thermostat and the modified Hamiltonian yields equations of motion that no longer require a scaling of the time step but still enables correct sampling of the canonical ensemble through MD simulation [72].

Barostats

Similarly to thermostats, barostats are used to maintain constant pressure during the simulation. In the following, the frequently used Parinello-Rahman barostat is introduced as an example.

At its core, the Parinello-Rahman barostat is similar to the Nose-Hoover thermostat, but this time an imaginary pressure bath couples to the original system instead of a heat bath. The resulting Hamiltonian is

$$H_{\text{Parinello-Rahman}} = \sum_i^N \left(\frac{\mathbf{p}_i^2}{2m_i} \right) + U(\mathbf{r}) + \sum_j \mathbf{P}_j j V + \sum_{j,k} \frac{1}{2} \mathbf{W}_{jk} \left(\frac{db_{jk}}{dt} \right). \tag{2.25}$$

Here, $\mathbf{b}$ is a matrix containing the box vectors and V is the volume of the simulation box, $\mathbf{P}$ is the instantaneous pressure tensor and $\mathbf{W}$ is the mass parameter matrix [79]. The vector of the simulation box b is coupled to the pressure bath via the relationships

$$\frac{db^2}{dt^2} = V\mathbf{W}^{-1}\mathbf{b}'^{-1}(\mathbf{P} - \mathbf{P}_{ref}), \tag{2.26}$$

where $\mathbf{P}_{ref}$ is the reference pressure [79].

2.3 Coarse-graining

Coarse-graining is the process of building a simplified model of a complex system. In particular, a target system is modeled at a low level of resolution, i.e. not all degrees of freedom of the system are treated explicitly. The goal is to keep essential features, while less important details are ignored or averaged over. In other words, the simplified model still has to maintain the correct physical behavior. The benefits of CG models are twofold: (1) Coarse-graining helps to put the essential features driving the emergent phenomena of interest into the spot light, as disturbing and unnecessary details are removed [80, 81]. (2) The reduced representation effectively accelerates the computer simulation of the system: Reducing the number of particles in a simulation reduces the number of required force calculations per simulation time step, i.e. a reduction of computational cost. Moreover, the CG configuration represents an average over an ensemble of microstates. As such, coarse-graining smoothens the energy landscape yielding "softer" interactions between the particles. Therefore, the time step applied in the numerical integration scheme can be increased [31, 32]. In addition, the smoothed energy landscape typically displays lower energy barriers along transition paths of metastable states and therefore accelerates dynamical processes [33, 34]. In conclusion, coarse-graining enables a faster exploration of configuration space and makes it possible to access length- and timescales that are not reachable with AA simulations.

At its core, coarse-graining consists of two steps: (1) Choose a low-resolution representation of the system and (2) build a CG force field in order to perform a computer simulation of the model. For the latter task, a wide range of different schemes have been developed and two schools of thoughts have been established, referred to as *bottom-up* and *top-down* approaches.

2.3.1 Representation

The basis of a CG model is the representation of the particles captured in the system. The most fundamental model relies on a quantum mechanical description. In this regard, a classical atomistic description is already a CG model based on ab initio considerations. However, coarse-graining typically refers to an even lower resolution description, where the CG sites, often called beads, represent multiple atoms. An illustrated of such a CG representation can be found in Fig. 2.2. Typically, beads are associated with specific types reflecting the physiochemical properties of the corresponding groups of atoms. Similar to AA representations, bonds between CG beads are introduced to capture the molecular topology. The representation of the CG system is crucial for the accuracy of the CG model, as it has to preserve essential features that are required to describe the phenomena of interest and to capture important slow and large amplitude motions of the system [82]. However, in many cases the mapping is based on the chemical intuition of the user, but more systematic methods have been developed recently [83, 84].

It is often required to not only specify the representation but also define a concrete mapping for the coordinates, i.e. a function $\mathbf{M}$ of the atomistic coordinates $\mathbf{r}$ to the coordinates of the CG beads $\mathbf{R}$. Typically, a linear mapping is chosen,

$$\mathbf{R}_I = \mathbf{M}_I(\mathbf{r}) = \sum_{i \in \psi_I} b_{iI} \mathbf{r}_i \tag{2.27}$$

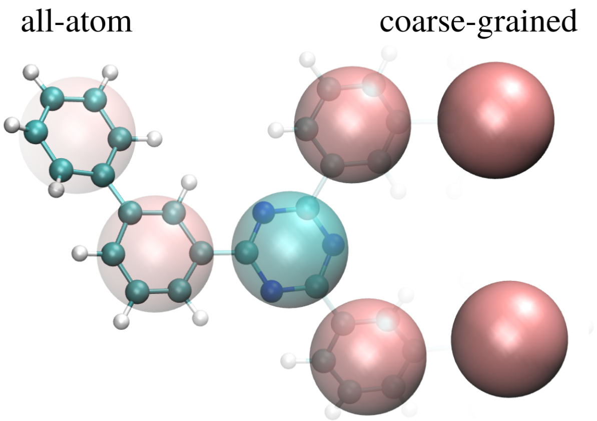

FIGURE 2.2: Illustration of a CG representation at the example of Tris-Meta-Biphenyl-Triazine. Atomistic representation (left) and CG representation (right). CG beads represent groups of atoms and are positioned at their center-of-mass.

where I and i are the indices of CG beads and atoms respectively, ψ_I is the set of atom indices that are associated with the CG bead I and b_{iI} are coefficients of the mapping. In many cases, the coordinate mapping reflects the center of mass geometry of the corresponding group of atoms, i.e. $b_{iI} = m_i / M_I$, where m_i is the mass of atom i and M_I is the total mass of all atoms associated with bead I.

2.3.2 Bottom-up Approach

Once the CG representation is chosen, the interactions between the beads have to be defined. The bottom-up strategy is an inductive approach that constructs a CG force field based on a more detailed model. In particular, bottom-up coarse-graining aims at reproducing energetic, thermodynamic or structural properties of the higher resolution system as closely as possible [85, 86, 87]. In general, the choice of the underlying fine-grained model is not bound to a specific resolution. A common choice is to use a classical atomistic model as a basis. In this case, the accuracy of the CG model depends on the quality of the fine-grained model, as the atomistic model itself is an approximation of the quantum mechanical description. Given a high-resolution model, statistical mechanics provides a framework to rigorously derive the force field for the CG system.

Consistency Criteria and the Many-Body Potential of Mean Force

Central to the bottom-up approach is the many-body potential of mean force (PMF). The PMF is an effective CG potential that includes energetic and entropic contributions [82]. The PMF can theoretically be derived exactly from the fine-grained potential $U_{\mathrm{AA}}(\mathbf{r})$ and the mapping $\mathbf{M}(\mathbf{r})$. The underlying criteria for the derivation is called consistency criteria and states that the equilibrium joint probability density $p_{CG}(\mathbf{R}, \mathbf{P})$ in phase space of the CG coordinates $\mathbf{R}$ and momenta $\mathbf{P}$ have to match the implied atomistic probability density $p_{AA}(\mathbf{R}, \mathbf{P})$ [88]. For simplicity, the following considerations are restricted to the configuration space, i.e. excluding momenta, such that the consistency criteria can be written as

$$p_{CG}(\mathbf{R}) = p_{AA}(\mathbf{R}), \tag{2.28}$$

where $p_{CG}(\mathbf{R})$ is is the equilibrium probability density for a configuration $\mathbf{R}$ in the canonical ensemble of the CG model

$$p_{CG}(\mathbf{R}) \propto \exp\Big[-\frac{U_{\mathrm{CG}}(\mathbf{R})}{k_B T}\Big] \tag{2.29}$$

and $p_{AA}(\mathbf{R})$ is the equilibrium probability density for a CG configuration $\mathbf{R}$ implied by the mapping $\mathbf{M}(\mathbf{r})$ expressed in terms of the fine-grained model

$$p_{AA}(\mathbf{R}) \propto \mathcal{Z}(\mathbf{R}) := \int \exp\Big[-\frac{U_{\mathrm{AA}}(\mathbf{r})}{k_B T}\Big]\delta(\mathbf{M}(\mathbf{r}) - \mathbf{R})d\mathbf{r}, \tag{2.30}$$

where δ is the Dirac delta distribution. Plugging Eq. 2.29 and 2.30 into 2.28 and reordering yields

$$U(\mathbf{R}) = -k_B T \ln(\mathcal{Z}(\mathbf{R})) + \text{const.} \tag{2.31}$$

Eq. 2.31 reveals that the many-body PMF is a projection of the free energy function onto the CG degrees of freedom. It assigns a weight to each CG configuration associated with the sum of all the Boltzmann weights for the corresponding atomistic configurations. Turning to the definition of the free energy F

$$F = U - TS = -k_B T \ln(\mathcal{Z}), \tag{2.32}$$

where U is the internal energy, makes it clear that the many-body PMF is not a regular potential, as it contains both, energetic as well as entropic contributions [88]. Moreover, as the name suggests, this potential generates the average atomistic forces associated with the atomistic configurations that map to the specific CG configuration [82].

Calculating the PMF provides significant challenges for most systems, since computing the free energy as a function of CG variables is computationally demanding and often unfeasible [89]. Typically, the PMF is approximated using molecular mechanics potentials as outlined in Sec. 2.2.2. However, the simple functional form of the interaction potentials might not provide an adequate basis set to approximate the PMF of CG systems [90]. While many-body interactions at the AA level can be captured approximately by renormalized pairwise terms, CG interactions often require more complex potential terms to correctly capture the effects of degrees of freedom that are averaged over [82]. A wide range of techniques have been developed to obtain tractable approximations of the many-body PMF that are still accurate enough to describe particular phenomena of interest and to perform simulations of the CG system.

Review of Bottom-Up Strategies

In this section, some popular bottom-up approaches to determine the CG potential are reviewed. Those approaches include structure based techniques, such as direct Boltzmann inversion (DBI) and iterative Boltzmann inversion (IBI), as well as variational approaches, like the multiscale coarse-graining approach (MS-CG) and the relative entropy (RE) framework.

Direct Boltzmann Inversion

The most simple approach to obtain an approximate CG potential is direct Boltzmann inversion (DBI) [57]. This approach aims at reproducing certain structural distributions computed from atomistic reference data that is mapped onto the CG degrees of freedom. The distribution functions for given interactions ξ are denoted with $p_\xi(x)$, where x is a scalar variable, such as pairwise distances, angles or dihedrals. The goal is to find the corresponding potential U_ξ that yields the desired distribution. Based on the assumption that the distribution functions for different mechanical variables x factorize, the probability distributions can be written as Boltzmann factors [57]

$$p_\xi(x) \propto \exp\Big[\frac{-U_\xi(x)}{k_B T}\Big]. \tag{2.33}$$

In addition, the corresponding volume elements for each distribution have to be taken into account, i.e. the Jacobian element $J_\xi(x)$ [57]. The potential can then directly be computed as

$$U_\xi(x) = -k_b T \ln\Big(\frac{p_\xi(x)}{J_\xi(x)}\Big). \tag{2.34}$$

Note that factorizing the probability distributions is a very severe approximation. Therefore, this approach yields accurate results only for interactions that can be regarded as isolated in the CG model [82]. However, in cases where the coupling between the interactions can not be ignored, DBI yields inaccurate potentials that do not correctly reproduce the structural distributions, as important cross-correlations are not taken into account.

Iterative Boltzmann Inversion

To improve the potentials derived with DBI, an iterative scheme can be applied. This scheme is called iterative Boltzmann inversion (IBI) and consists of the following steps [91, 92]: (1) Initial potentials are derived using DBI. (2) Use the derived CG potentials in a MD simulations to obtain the corresponding structural distributions $p_\xi(x|U)$. (3) Update the CG potentials via

$$U_{\xi,\text{new}}(x) = U_{\xi,\text{old}}(x) - k_B T \ln\Big(\frac{p_\xi(x)}{p_\xi(x|U)}\Big). \tag{2.35}$$

Steps (2) and (3) are repeated until the potentials converge. Despite its simplicity, IBI has become very popular and is used especially for complex liquids and polymers. While IBI still treads every interaction ξ separately, it implicitly accounts for correlations between the interactions through the iterative scheme [82]. However, convergence of the potentials is not guaranteed [82].

Multiscale Coarse-graining

Beside approaches that focus on reproducing certain structural properties, variational approaches can be applied to approximate the many-body PMF. The multiscale coarse-graining (MS-CG) method is a variational approach, which introduces a force-matching functional [93, 60, 94]

$$\chi^2[\mathbf{F}] = \frac{1}{3N}\left\langle \sum_{I=1}^{N} |\mathbf{F}_I(\mathbf{M}(\mathbf{r})) - \mathbf{f}_I(\mathbf{r})|^2 \right\rangle_{\text{AA}}, \tag{2.36}$$

where $\mathbf{F}_I(\mathbf{M}(\mathbf{r}))$ is the force acting on the CG site I mapped from the atomistic configuration implied by the trial potential of the CG model and $\mathbf{f}_I(\mathbf{r})$ is the net force acting the group of atoms associated with site I. The angular brackets with subscript AA denotes the atomistic canonical ensemble average, which is typically approximated using trajectories obtained in a simulation. The MS-CG method states that the CG potential yielding the best approximation of the average net atomistic forces should be used and therefore $\chi^2[\mathbf{F}]$ has to be minimized. Indeed, the functional $\chi^2[\mathbf{F}]$ has a unique global minimum given by the actual many-body PMF [82]. In practice, the CG force F is expressed as a linear combination of basis functions leading to a coupled system of linear equations that can be solved directly [82]. As such, the choice of an adequate basis set is crucial for the accuracy of this method.

Relative Entropy

Another variational approach is based on the relative entropy, also known as Kullback-Leibler divergence, which is widely used as an asymmetric distance metric between probability distributions [95, 96, 97, 98]. Applied to the coarse-graining problem, the relative entropy can be written

$$S_{\text{rel}} = \int p_{\text{AA}}(\mathbf{r}) \ln\left(\frac{p_{\text{AA}(\mathbf{r})}}{p_{\text{CG}}(\mathbf{M}(\mathbf{r}))}\right) d\mathbf{r} + \langle S_{\text{map}} \rangle_{\text{AA}}, \tag{2.37}$$

where

$$S_{\text{map}}(\mathbf{R}) = \ln \int \delta[\mathbf{M}(\mathbf{r}) - \mathbf{R}] d\mathbf{r} \tag{2.38}$$

is the mapping entropy, which accounts for the degeneracy of the mapping $\mathbf{M}$, i.e. a single CG configuration corresponds to multiple atomistic configurations. The relative entropy S_{rel} can be interpreted as a measure for the loss of information when changing from an atomistic to a CG description [96]. Moreover, it is related to many different coarse-graining errors and vanishes only if $p_{\text{CG}}(\mathbf{r}) \propto p_{\text{AA}}(\mathbf{r})$ [97].

Inserting the distributions known for the canonical ensemble, i.e.

$$p_{\mathrm{CG}}(\mathbf{R}) = \mathcal{Z}_{\mathrm{CG}}^{-1}\exp\Big[-\frac{U_{\mathrm{CG}}(\mathbf{R})}{k_B T}\Big] \quad \text{and} \quad p_{\mathrm{AA}}(\mathbf{r}) = \mathcal{Z}_{\mathrm{AA}}^{-1}\exp\Big[-\frac{U_{\mathrm{AA}}(\mathbf{r})}{k_B T}\Big], \tag{2.39}$$

into Eq. 2.37 yields

$$S_{\mathrm{rel}} = \frac{\langle U_{\mathrm{CG}} - U_{\mathrm{AA}}\rangle_{\mathrm{AA}}}{k_B T} - \frac{F_{\mathrm{CG}} - F_{\mathrm{AA}}}{k_B T} + \langle S_{\mathrm{map}}\rangle_{\mathrm{AA}}, \tag{2.40}$$

where $F = -k_b T\,\ln(\mathcal{Z})$ is the free energy. In order to optimize the parameters λ of the CG potential U_{CG} the derivative of the relative entropy can be used

$$\frac{\partial S_{\mathrm{rel}}}{\partial \lambda} = \frac{1}{k_B T}\Big[\Big\langle \frac{\partial U_{\mathrm{CG}}}{\partial \lambda}\Big\rangle_{\mathrm{AA}} - \Big\langle \frac{\partial U_{\mathrm{CG}}}{\partial \lambda}\Big\rangle_{\mathrm{CG}}\Big], \tag{2.41}$$

i.e. the derivatives of U_{CG} with respect to its parameters have to average to the same value for both, the atomistic and the CG ensemble [96]. This approach reproduces the expectation values for every observable, such as distances or angles, that are included in the CG potential, i.e. the relative entropy framework can be used to generate potentials that capture the correct structural distributions [96]. Numerical minimization of the relative entropy is typically achieved using advanced algorithms, such as the Newton-Raphson method [96].

2.3.3 Top-down Approach

While bottom-up models are build upon higher resolution models, top-down coarse-graining follows a deductive philosophy: Top-down CG models are designed to study the consequences upon application of general rules [99, 100, 80]. Such rules are typically inferred from universal physical principles or constructed to reproduce specific phenomena that have been observed experimentally [82]. In most cases, simple potentials are applied that are defined by relatively few parameters, which can systematically be varied to study the consequences of certain aspects of the model [101, 102]. In particular, top-down models can be chemically-specific or generic: Chemically-specific models aim at reproducing certain properties of a target system, such as density, interfacial tension or partitioning of compounds between aqueous and hydrophobic environments [103, 104, 105]. On the other hand, generic models are designed without relating to any particular system [106, 107, 108]. Typically, generic top-down models address large-scale phenomena at a low resolution. As such, they lack chemical details and it is not straightforward to relate them to higher resolution models.

Review of Top-down Models

This section reviews some popular top-down force fields. In particular, the hydrophobic-polar (HP) protein model, the Kremer-Grest (KG) polymer model and the Martini model for biomolecular systems are presented.

Hydrophobic-polar Protein Model

The hydrophobic-polar (HP) protein model is a highly simplified model to study protein folds [109]. The HP model is a lattice model that represents proteins as two or three-dimensional self-avoiding walks. It assumes that the hydrophobic interaction between amino acid residues is the driving force for proteins folding into their native states. In particular, the model represents proteins as sequences of hydrophobic (H) and polar (P) residues. The hydrophobic effect is imitated by assigning a negative weight to interactions between adjacent, non-covalently bound H residues in order to stabilize the contact. The native structure of a protein is identified as the conformation that maximizes the number of contacts between hydrophobic residues. Despite its simplicity, the HP model has been a corner stone for lattice models and lead to the development of more advanced methods that are able to determine minimum energetic states for long protein sequences close to experimentally observed conformations [110, 111].

Kremer-Grest Polymer Model

The Kremer-Grest (KG) model is a top-down model widely used to study generic polymer properties [112, 113]. The model represents polymers as chains of beads connected via non-linear springs. The spring potential is tuned such that crossing of two polymer chains is avoided in order to correctly simulate the dynamics of polymer melts, especially entanglement effects of long chains. More specifically, the potential for bonded beads is given by the finite-extensible-nonlinear spring (FENE) potential

$$U_{\text{FENE}}(r) = -15 k_B T \left(\frac{R}{\sigma}\right)^2 \ln\left[1 - \left(\frac{r}{R}\right)^2\right], \tag{2.42}$$

where r is the distance between the two bonded beads, σ is the bead diameter and R defines the distance where the potential divergences. A typical choice is to set $R = 1.5\sigma$. Additionally, the beads interact through a truncated and shifted Lennard-Jones potential, which is purely repulsive

$$U_{\text{WCA}}(r) = \begin{cases} 4 k_B T \left[\left(\frac{\sigma}{r}\right)^{-12} - \frac{\sigma}{r}\right)^{-6} + \frac{1}{4}\right] & , \text{ if } r < 2^{1/6}\sigma \\ 0 & , \text{ otherwise} \end{cases} . \tag{2.43}$$

In order to vary the stiffness of the chains, an additional bending potential can be introduced

$$U_{\text{bend}}(\Theta) = \kappa k_B T (1 - \cos(\Theta)), \tag{2.44}$$

where Θ is the the bond angle and κ defines the stiffness [114].

While the KG model is a generic model to study universal phenomena of polymer melts, the stiffness of the chains can be used to relate the model to real polymers [115, 116]. To this end, simulated and real polymer melts can be linked via their Kuhn length, i.e. the scale indicating the crossover from a chemistry-specific to a universal random-walk like behavior (see Sec. 7.2.1 for a more detailed explanation).

Martini Force Field

The Martini force field is a generic CG potential for a wide range of soft matter systems. It was developed with an emphasize on biomolecules and its various applications include lipid membranes, proteins, sugars and nucleotides [117, 118, 119, 120]. The parameterization of the Martini force field incorporates both coarse-graining philosophies, a top-down approach for the non-bonded interactions and a bottom-up approach for the bonded interactions. While the non-bonded parameters of the model are tuned to reproduce experimental partitioning free energies of water-alkane mixtures, the bonded interactions are optimized to capture the correct conformational distributions of atomistic reference data [121].

Central to the design of the Martini model is a robust transferability across soft matter systems. For this reason, the model is based on modular building blocks, i e. CG beads, and introduces rules for the mapping from groups of atoms to the beads. On average, four heavy atoms and their associated hydrogens are represented by a single CG site. The four main building blocks are denoted with charged (Q), polar (P), nonpolar (N), and apolar (C). Each of the main bead types have further subtypes distinguishing either their hydrogen bonding capability (Q and N types) or their degree of polarity (P and C types). The assignment of the specific bead types is based on the hydrophobicity, i.e. the water/organic partition free energy, of the corresponding group of atoms.

Despite its wide use and robust transferability, the Martini model also has its limitations. A major drawback of the Martini model is a less accurate reproduction of structural features for particular systems, which is reasonable for a force field parameterized with an emphasis on transferability. For example, the model does not include size-dependent Lennard-Jones parameters which may lead to artifacts, such as increased barriers in dimerization profiles [122].

2.4 Reverse-mapping

Reducing the resolution is just one side of the multiscale philosophy. In order to close the loop, a reverse-mapping to go the other direction is required as well. Such a reverse-mapping, also referred to as *backmapping*, can be regarded as a magnifying glass to zoom into the molecular system.

CG models can be used to study processes on large length- and long timescales that would have been too costly and time consuming to be reached with higher resolution models. However, while CG models lack the accuracy and details of atomistic simulations, atomistic details are often required for one or more of the following reasons: (1) To rigorously analyze the simulation results on a local scale [40, 41, 42, 43],

(2) to enable a direct comparison to experimental data, for example obtained with spectroscopic methods [44], (3) to serve as starting point for further high-resolution simulations [45, 41], or (4) to asses the stability and accuracy of the obtained CG structures [45]. Therefore, reverse-mapping is an integral part of MM. In particular, details are reintroduced along the CG degrees of freedom, i.e. large-scale characteristics of the system are retained upon backmapping. As such, reverse-mapping becomes feasible, since the reintroduced degrees of freedom have to be equilibrated only locally. In summary, combining coarse-graining and reverse-mapping is a powerful tool to obtain well equilibrated molecular trajectories at a high resolution for processes that require to consider large length- and long timescales.

2.4.1 The Challenges of Reintroducing Degrees of Freedom

While mapping from a higher to a lower resolution is typically straightforward, the opposite direction is more challenging. Formally, let $\{\mathbf{A}_I = (\mathbf{R}_I, \mathbf{C}_I) | I = 1, \ldots, N\}$ denote the set of N CG beads. Each bead has position $\mathbf{R}_I$ and an associated type $\mathbf{C}_I$. The type $\mathbf{C}_I$ reflects various attributes, such as the bead mass, the connectivity or associated force field parameters. Similarly, let $\{\mathbf{a}_I = (\mathbf{r}_i, \mathbf{c}_i) | i = 1, \ldots, n\}$ denote the set of n atoms, with position $\mathbf{r}_i$ and types $\mathbf{c}_i$.

A backmapping function ϕ takes the CG information $\mathbf{A} = (\mathbf{A}_1, \ldots, \mathbf{A}_N)$ as well as the target atom types $\mathbf{c} = (\mathbf{c}_1, \ldots, \mathbf{c}_n)$ as input and generates a set of coordinates $\mathbf{r} = (\mathbf{r}_1, \ldots, \mathbf{r}_n)$,

$$\phi(\mathbf{A}, \mathbf{c}) = \mathbf{r}. \tag{2.45}$$

Deriving the function ϕ is not a trivial task, as it is constrained by two important aspects: (1) The mapping has to be consistent, i.e. the missing degrees of freedom $\mathbf{r}$ have to be reinserted along the CG degrees of freedom $\mathbf{R}$. In other words, applying the CG mapping $\mathbf{M}$ to the backmapped structure $\phi(\mathbf{A}, \mathbf{c})$ has to yield the original CG structure,

$$\mathbf{M}(\phi(\mathbf{A}, \mathbf{c})) = \mathbf{R}. \tag{2.46}$$

(2) The mapping is not unique, since the reduced resolution implies that many atomistic structures can map to the same CG configuration. As a consequence, a single CG structure $\mathbf{R}$ will correspond to an ensemble of atomistic microstates $\{\mathbf{r} | \mathbf{M}(\mathbf{r}) = \mathbf{R}\}$. Therefore, strictly speaking, the CG mapping is not invertible.

In order to take the aforementioned aspects into account, the backmapping problem is expressed as a joint conditional probability distribution $p_\phi(\mathbf{r}|\mathbf{A}, \mathbf{c})$ that assigns a statistical weight to each atomistic configuration $\mathbf{r}$ given the CG information $\mathbf{A}$ as well as the atomistic attributes $\mathbf{c}$ of the target system. Ideally, the CG model yields the Boltzmann distribution expressed in the CG degrees of freedom, i.e. the many-body PMF is reproduced perfectly. Consequently, an ideal backmapping scheme also reinserts atomistic details with the correct statistical weight, i.e.

$$p_\phi(\mathbf{r}|\mathbf{A},\mathbf{c}) = \begin{cases} \propto \exp\Big(-\frac{U(\mathbf{r})}{k_B T}\Big) & \text{if } \tilde{\mathbf{M}}(\mathbf{r},\mathbf{c}) = \mathbf{A} \\ 0 & \text{if } \tilde{\mathbf{M}}(\mathbf{r},\mathbf{c}) \neq \mathbf{A} \end{cases}, \tag{2.47}$$

where $\tilde{\mathbf{M}}$ is used as an extended coarse-graining mapping function that includes both, the coordinates as well as the types of the atoms and beads, respectively.

2.4.2 Review of Existing Approaches

Backmapping is widely used in the MM community and several approaches exist. Traditional methods include fragment-based and generic approaches that follow a similar strategy: An initial structure is generated using some heuristics and then refined via energy minimization (EM) and short runs of MD simulations. More recently, approaches based on machine learning (ML) have been used for reconstruction as well that do not rely on EM and MD.

Fragment-based Approaches

Fragment-based backmapping superimposes several trial atomistic fragments onto associated CG sites [37, 48, 44, 49]. Typically, rigid rotation and translation is used to optimize the orientation of the given fragment with respect to some geometric or energetic properties. The trial fragments are usually drawn from a presampled library. The result is an initial atomistic structure that is most likely not representative for the Boltzmann distribution. In particular, a simple projection of fragments onto the CG sites usually leads to overlaps between reconstructed atoms and distorted bonded structures. It is therefore necessary to relax the initial structure by means of energy minimization deploying the atomistic force field. Subsequently, the relaxed structure has to be equilibrated using MD simulation in order to recover the correct statistical weights of the new degrees of freedom. When the overall equilibration and diffusion is rather slow compared to the equilibration of local features, the equilibration process is straightforward. In other cases, restraints have to be introduced to prevent the reconstructed atoms to drift too far away from the center of their associated CG site [123, 124, 125]. To this end, an additional potential can be applied that couples the atomistic degrees of freedom to the CG degrees of freedom

$$U_{\text{restr}}(\mathbf{r},\mathbf{R}) = b(\mathbf{M}(\mathbf{r}) - \mathbf{R})^2, \tag{2.48}$$

where the prefactor b is used to scale the restraining potential.

Generic Approaches

Generic backmapping approaches are similar to fragment-based approaches, but differ in the way the initial atomistic structure is derived. Generic schemes do not rely on presampled fragments but project the atomistic degrees of freedom onto the CG structure using general rules. In the most basic version, the atoms are randomly placed close to their associated CG bead, whereas more sophisticated approaches rely on geometric rules to place the atoms [46, 47]. The resulting initial atomistic structure is typically even more distorted as in the fragment-based approach. Therefore, the subsequent energy minimization and equilibration procedures have to be performed more carefully and typically involve multiple stages, where the interaction potentials are gradually switched on.

Machine Learning Approaches

During the course of this PhD several ML approaches have been published to tackle the backmapping problem. The increased interest of integrating ML techniques to the field of MM is encouraging and emphasizes the importance of the present work.

Wang *et al.* utilized a variational autoencoder (Sec. 3.4.2) and treated the CG degrees of freedom as latent variables (Sec. 3.2.2) [126]. Their framework unifies the task of learning the CG variables, parameterizing the CG force field and decoding back to atomistic resolution. In contrast to standard variational autoencoder, where the latent distribution is regularized to resemble a Gaussian distribution, the proposed coarse-graining autoencoder utilizes a force-matching functional for regularization. The approach is tested for gas-phase molecules and bulk simulations of alkanes. In all cases, the method was able to reproduce a reasonably accurate structural correlation function for decoded configurations. However, the deterministic decoder trained with mean-square error as reconstruction loss leads unavoidably to a loss of mapping entropy. Therefore, the decoder has learned to generate a mean reconstruction of an ensemble of microstates, which limits structural fidelity, as can be seen for the distribution of bond lengths. As a remedy, a probabilistic decoder is suggested to improve the model and to yield higher fidelity reconstructions.

Li *et al.* utilized a convolutional conditional generative adversarial network (Sec. 3.4.3) for the reconstruction of cis-1,4-polyisoprene melts from a CG representation [127]. This approach is similar in spirit to the method proposed in this thesis (Sec. 4), but is based on an image representation where XYZ components of vectors are converted into red–green–blue (RGB) values. While being computational efficient, the method does not fully take the local environment of the polymer chains into account. As a consequence, steric overlaps are observed and demand for further relaxation via energy minimization.

An *et al.* used several ML approaches including artificial neural networks, k-nearest neighbor, Gaussian process regression and random forest to built regression models for the backmapping task [128]. The regression was performed for small molecules in vacuum and the coordinates of the CG and atomistic structures were directly used as input and output representations for the models. The best performance was achieved by an artificial neural network. However, backmapping of the alkane hexane provided significant challenges for all deployed models.

Chapter 3

Machine Learning

Machine Learning is a prominent subfield of *Artificial Intelligence (AI)* and is already used in a wide range of applications, such as computer vision, speech recognition or medical image analysis [6, 7, 8]. It is a study of computer algorithms that use data to construct statistical models trained to perform specific tasks. The models improve their performance automatically by learning from examples, instead of relying on static program instructions. Importantly, learning in this context aims at extracting patterns or rules from the training data that generalize rather than simply memorizing specific examples.

Recently, ML is gaining significant attention in many fields of modern science, especially computational chemistry and particle physics [9, 10, 11]. Beside the massive increase in computational power, the growing interest for ML algorithms in those research areas is fueled by the availability of large data sets [129, 130, 131]. The massive amount of raw data collected in experiments or computer simulations demands for efficient algorithms to process and analyze it. Self-learning algorithms that improve their performance with increased data set size are therefore very appealing. This is especially true when the data is high-dimensional, as ML algorithms can be used to spot complex patterns or to reduce the dimensionality for further processing.

Data analysis is also a hallmark of classical statistics. Indeed, ML and classical statistics are related and many techniques and concepts used in ML have their origin in physics, such as variational methods, simulated annealing or Monte-Carlo methods [132]. A famous example is the Boltzmann Machine, which has a direct analogy to a spin glass model known from statistical mechanics [133]. However, despite their similarities, ML and classical statistics differ in their general philosophy: Central to most ML approaches is the transferability of the model, i.e. ML algorithms are typically designed to make predictions for new observations that are not part of the current data set [132]. On the other hand, methods of classical statistics are more concerned with estimation problems, i.e. to discover dependencies between variables of the current data set.

In this thesis, a ML model is used to predict fine-grained states of a molecular system based on coarse-grained states. This chapter introduces important ML concepts required to build such a ML-based reverse-mapping model. While ML is an umbrella term for a wide spectrum of algorithms, such as Kernel methods, decision trees or deep neural networks (DNNs), it is out of scope of this chapter to cover all branches in this field. The interested reader is therefore referred to one of the many excellent books available [134, 135, 136]. Instead, this chapter focuses on generative modeling using DNNs, which is the main method used in this thesis. The rest of this chapter is organized as follows: Firstly, basics of ML are introduced and some important aspects of high-dimensional systems are described. Secondly, DNNs are explained in detail and finally, concepts of generative modeling are outlined.

3.1 Basics

Most problems in ML are tackled by a common scheme: At first, a *data set* $\mathcal{D} = \{\mathbf{x}\}$ is collected that consists of a set of independent and identically distributed variables $\mathbf{x}$ sampled from a distribution $\mathcal{X}$, which is typically high-dimensional and intractable. In general, $\mathbf{x} \in \mathrm{R}^D$ is a vector with dimension D or a tensor of higher rank, such as an image. Secondly, a *model* $f_\Theta(\mathbf{x}) := f(\mathbf{x}; \Theta) = \hat{\mathbf{y}}$ is introduced as a function

$$f_\Theta : \mathrm{R}^D \to \mathrm{R}^S, \tag{3.1}$$

with parameters Θ that maps the input $\mathbf{x}$ to some output $\hat{\mathbf{y}} \in \mathrm{R}^S$ with dimension S. The last ingredient required to train the model is a *cost function* $\mathcal{C}(f_\Theta(\mathbf{x}))$

$$\mathcal{C} : \mathrm{R}^S \to \mathrm{R} \tag{3.2}$$

that maps the output $\mathbf{y}$ of $f_\Theta(\mathbf{x})$ to a real number representing the error the model has made on $\mathbf{x}$. That is, $\mathcal{C}$ is used to judge the performance of the model. During training, the parameters of the model Θ are tuned such that the cost function is minimized aiming at discovering the optimal set of parameters Θ^*.

ML algorithms split into *supervised* and *unsupervised* learning approaches. The supervised learning approach deploys labeled data $\mathcal{D} = \{(\mathbf{x}, \mathbf{y})\}$ that consists of pairs of input variables $\mathbf{x}$ and associated output variables $\mathbf{y}$. Consequently, the model $f_\Theta(\mathbf{x})$ is trained to predict the desired output $\mathbf{y}$ and the cost function $\mathcal{C}(\mathbf{y}, f_\Theta(\mathbf{x}))$ becomes a function of both, the output of $f_\Theta(\mathbf{x})$ as well as the actual label $\mathbf{y}$. For discrete outputs $\mathbf{y}$, the task becomes classification, while continues variables refer to regression [137]. The unsupervised approach does not use labels explicitly to train the ML model, but aims at learning the underlying structure of the data instead. Examples for the unsupervised approach include generative modeling, clustering and dimensionality reduction [138, 139, 140]. However, in some ML algorithms the distinction between supervised and unsupervised learning becomes fuzzy. An example of such *semi-supervised* algorithms is the generative adversarial approach (Sec. 3.4.3), which will be an important ML algorithm throughout this thesis.

3.1.1 Interpretation of Probability: Bayesian vs Frequentist

In the field of statistical inference two different interpretations of probability can be found, known as the *Bayesian* and the *frequentist* paradigm. The debate over the different views of probability is going on for more than 250 years and dates back to a publication of Thomas Bayes titled "An Essay towards solving a Problem in the Doctrine of Chances" [141].

In the frequentist view, probability is objective and is only discussed for well defined random-experiments [142]. In particular, probability is defined as the relative frequency of an event with which it occurs in many trials. Since relative frequencies can vary in different experiments, true probabilities only exist in the limit of infinite trials where the difference in the relative frequencies diminish. In practice, frequentists follow a deductive approach: They introduce a model, i.e. a point in parameter space Θ, and ask for its consistency with observed data, which is considered as the random variable. As such, the focus is set on the *likelihood* $p(\mathcal{D}|\Theta)$ of observing the data given the model parameters. Consequently, frequentists tune model parameters such that the likelihood is maximized.

In the Bayesian view, probability is subjective and quantifies uncertainty or the degree of personal belief [143]. Bayesians do not seek a point estimate for the parameters of a model but a distribution $p(\Theta|\mathcal{D})$, called *posterior distribution*. As such, the parameters Θ are treated as random variables and the likelihood for a model explaining the observed data is of interest. Importantly, belief in the model parameters prior to observing any data can be expressed in a *prior probability* $p(\Theta)$. In practice, the collected data is used to update the belief in the model parameters deploying the likelihood $p(\mathcal{D}|\Theta)$ and the prior probability $p(\Theta)$. Applying Bayes theorem, the posterior can be computed as

$$p(\Theta|\mathcal{D}) = \frac{p(\mathcal{D}|\Theta)p(\Theta)}{p(\mathcal{D})}, \tag{3.3}$$

where $p(\mathcal{D}) = \int p(\mathcal{D}|\Theta)p(\Theta)d\Theta$ is the *evidence* or marginal likelihood [143]. However, computing the posterior distribution of the model parameters is computationally demanding and often intractable, because of the evidence term. In practice, Bayesians perform a maximum a posteriori (MAP) estimation, which bypasses the cumbersome computation of the posterior distribution, but tries to find a point estimate of the parameters that maximize the posterior distribution instead.

While a distinction between the Bayesian and the frequentist probability interpretation is instructive, it is not always possible to strictly assign a given method to one of both paradigms [144]. For example, MAP estimation can be treated as a maximum likelihood approach, when a uniform prior distribution $p(\Theta)$ is assumed. In practice, both interpretations are routinely used and many ML algorithms incorporate aspects of both ideas.

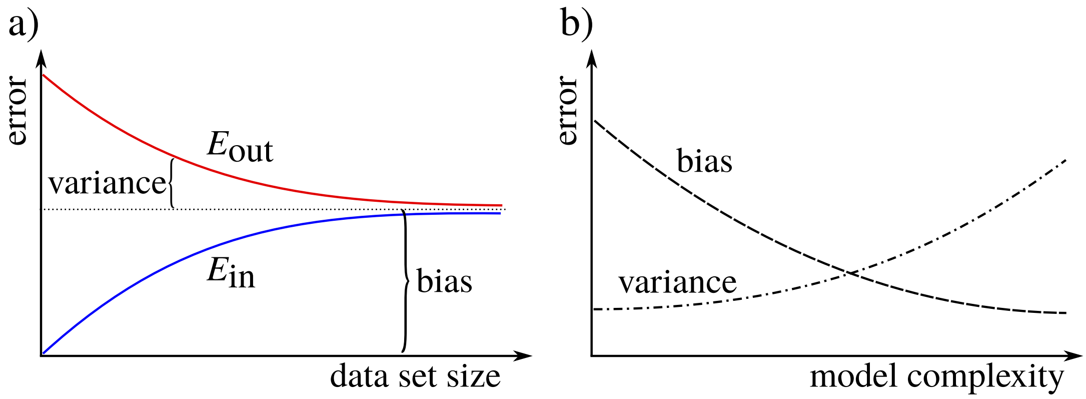

FIGURE 3.1: a) Illustration of the typical dependence of the in-distribution E_{in} and out-of-distribution E_{out} error with respect to the training set size. E_{out} is composed of two terms: the bias and variance. Initial drop of the error is omitted in the figure. b) Bias–Variance tradeoff and model complexity. The out-of-distribution error E_{out} is plotted as a function of the model complexity for a training data set of fixed size. While the bias decreases with model complexity, the variance increases with model complexity.

3.1.2 Bias-Variance Tradeoff

It is common practice to split the data set $\mathcal{D}$ randomly into two exclusive subsets: The training set $\mathcal{D}_{\text{train}}$ and the test set $\mathcal{D}_{\text{test}}$. Typically, the training set $\mathcal{D}_{\text{train}}$ contains the majority of the data. During training, the parameters Θ of the model f_Θ are tuned to minimize the cost function $\mathcal{C}$ evaluated on the training set $\mathcal{D}_{\text{train}}$ only. The error on the training set

$$E_{\text{in}} = \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{\text{train}}} \mathcal{C}(\mathbf{y}, f_\Theta(\mathbf{x})) \tag{3.4}$$

is called the *in-distribution error*. After training, the performance of the model is evaluated computing the cost function with respect to the test set,

$$E_{\text{out}} = \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{\text{test}}} \mathcal{C}(\mathbf{y}, f_\Theta(\mathbf{x})), \tag{3.5}$$

which is called the *out-of-distribution error*. This procedure is known as *cross-validation* and its purpose is to find an unbiased estimate for the predictive performance of the model. In most cases, the out-of-distribution error is greater than the in-distribution error [132].

The general relationship between the training error E_{in} and the generalization error E_{out} is summarized in Fig. 3.1 a), where both errors are plotted as a function of the training set size. The following consideration is based on the assumption that the underlying data distribution is sufficiently complex, such that the model will not be able to perfectly reproduce it. Therefore, after an initial drop (excluded in the figure), the in-distribution error E_{in} will increase with the amount of training data, as the model is not powerful enough to capture all the regularities of the training set accurately. On the other hand, the out-of-distribution error E_{out} will decrease with the training set size, as the sampling noise decreases and the training set becomes more representative of the true data distribution. Consequently, both errors will converge to the same value in the limit of infinite training set size [132]. The error in the limit of infinite data is called *bias* and the fluctuation of E_{out} due to a limited training set size is referred to as *variance* of the model.

The out-of-distribution error E_{out} is a combination of both, the bias and the variance. An exact decomposition for the expectation of E_{out} can be derived for a regression model trained with mean-square-error (MSE) [132]: Consider a data set $\mathcal{D} = \{(\mathbf{x}, \mathbf{y})\}$ sampled from a noisy model

$$\mathbf{y} = f(\mathbf{x}) + \epsilon, \tag{3.6}$$

where ϵ is normally distributed with zero mean and standard deviation σ_ϵ. The model parameters $\Theta^*_{\mathcal{D}}$ are obtained by minimizing the squared error for the data set $\mathcal{D}$. Since $\mathcal{D}$ is finite, the parameters $\Theta^*_{\mathcal{D}}$ found will vary for different data sets. Denoting the expectation over all possible data sets with $\mathrm{E}_{\mathcal{D}}$, i.e. the asymptotic value in the limit of infinite data, and the expectation over the noise with E_ϵ, the expected out-out-sample error $\mathrm{E}_{\mathcal{D},\epsilon}[E_{out}]$ can be decomposed as

$$\begin{aligned} \mathrm{E}_{\mathcal{D},\epsilon}[E_{out}] &= \mathrm{E}_{\mathcal{D},\epsilon}\Big[\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{\text{test}}} (\mathbf{y} - f_{\Theta^*_{\mathcal{D}}}(\mathbf{x}))^2\Big] & (3.7) \\ &= \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{\text{test}}} \underbrace{\sigma^2_\epsilon}_{noise} + \underbrace{\Big(f(\mathbf{x}) - \mathrm{E}_{\mathcal{D}}[f_{\Theta^*_{\mathcal{D}}}(\mathbf{x})]\Big)^2}_{bias^2} + \underbrace{\mathrm{E}_{\mathcal{D}}\Big[\Big(f_{\Theta^*_{\mathcal{D}}}(\mathbf{x}) - \mathrm{E}_{\mathcal{D}}[f_{\Theta^*_{\mathcal{D}}}(\mathbf{x})]\Big)^2\Big]}_{variance}. & (3.8) \end{aligned}$$

While trained to minimize the in-distribution error, the ultimate goal for a predictor is to achieve a low out-of-distribution error. All three terms in Eq. 3.7 are positive and therefore each of them has to be minimized. The noise term is not affected by the model and therefore irreducible. The other two terms, the bias and the variance, are reducible. Unfortunately, minimizing both of them simultaneously poses challenges, known as the *bias-variance tradeoff* [132]. The bias term

$$bias^2 = \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{out}} \Big(f(\mathbf{x}) - \mathrm{E}_{\mathcal{D}}[f_{\Theta^*_{\mathcal{D}}}(\mathbf{x})]\Big)^2 \tag{3.9}$$

is the deviation of the models estimate in the limit of infinite data from the true value. A model with high bias is said to *underfit* the data. On the other hand, the variance

$$variance = \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{out}} \mathrm{E}_{\mathcal{D}}\Big[\Big(f_{\Theta^*_{\mathcal{D}}}(\mathbf{x}) - \mathrm{E}_{\mathcal{D}}[f_{\Theta^*_{\mathcal{D}}}(\mathbf{x})]\Big)^2\Big] \tag{3.10}$$

describes how much the model is expected to fluctuate around the ideal estimate in the infinite data limit. In particular, the fluctuations occur due to finite size effects of the training set. A model with high variance is said to *overfit* the data set.

As illustrated in Fig. 3.1 b), both errors depend on the complexity of the model, which is related to the number of degrees of freedom in the model, i.e. the number of parameters. However, the bias always decreases with the model complexity while the variance might increase instead [132]. Intuitively, a very complex model does not only learn the regularities of the training data but also the sample noise. Essentially, if the training set becomes too small, a complex model can simply remember every detail of the training examples. Therefore, the generalization will suffer from overemphasizing particular details in the training set leading to a large discrepancy between the in-distribution and out-of-distribution error. Consequently, a less-complex model with low variance but high bias can be superior in cases where the data set size is small [132].

The bias-variance tradeoff highlights the importance of large data sets for ML, as it offers the ability to deploy more complex models. Beside increasing the training set size, generalization can be improved using *regularization* techniques. Regularization enforces constraints on the model and thereby limits the functional space of the model. In other words, learning of an overly complex function is discouraged in order to avoid overfitting. In the frequentist approach, regularization is typically enforced by additional terms in the cost function that penalize overspecialized parameter settings. Common examples include $L1$ and $L2$ terms, where the absolute value or the square value of the parameters are penalized. In the Bayesian view, regularization refers to a prior distribution on the model parameters, i.e. less complex models have a higher probability. In practice, both approaches are closely related. For example, if a Gaussian prior distribution is assumed, MAP is equal to the maximum likelihood approach with $L2$ regularization, whereas MAP estimation with a Laplacian prior distribution is equal to maximum likelihood with $L1$ regularization.

3.2 High-Dimensional Data

Big data has become a hallmark of modern ML. This development is fueled to a great extent by a continues increase of computational power that makes it possible to collect extremely large data sets $\mathcal{D} = \{\mathbf{x}\}$. However, many data sets do not only contain a large number of observations $\mathbf{x} = (x_1, x_2, .., x_D)$ but the observations also become high-dimensional, i.e. D becomes very large. Various phenomena that occur when dealing with high-dimensional spaces are counter intuitive and provide challenges as well as opportunities for modern ML.

3.2.1 Curse and Blessing of Dimensionality

Depending on the point of view, phenomena related to the high dimensionality of the data are referred to as *curse* or *blessing of dimensionality* [145]. Beside a performance degradation in terms of speed and efficiency of algorithms in high-dimensional spaces, more fundamental challenges appear. A common theme of such problems is the sparsity of available data [145]. As an example, consider a set of points within a D-dimensional unit cube with a fixed spacing of $\frac{1}{10}$ along each axis of the Cartesian grid. The number of points n will grow exponentially with the dimension, i.e. $n = 10^D$. Conversely, increasing the dimension for a fixed number of points lets the distances between those points grow exponentially, ultimately leading to an almost empty space. Therefore, approximating a function in a high-dimensional space becomes intractable, as there is never enough data to support the result [145, 146].

Another challenge in high-dimensional spaces is the choice of an appropriate metric. The Euclidian distance is a well suited distance measure in the three-dimensional physical space, but in higher dimensions the distances grow more and more alike. It can be shown, that for a wide range of distributions and distance functions the ratio of distances of the nearest and farthest neighbors to a given target is almost unity [147]. Therefore, the distribution of distances tend to concentrate and lose contrast. For this reason, the concept of nearest neighbor becomes meaningless and similarity search ill-posed [147, 146].

While the previously mentioned challenges are examples for the curse of dimensionality, it might also be regarded as a blessing. In many cases, the high dimensionality can lead to simple laws and reduces the impact of fluctuations. A famous example is the central limit theorem: The joint effect of random phenomena tends toward a normal distribution if the number of such phenomena is large. For this reason, the normal distribution is ubiquitous. Moreover, the benefits of high-dimensional systems are well known in statistical mechanics. In the thermodynamic limit, where the number of particles tends to infinity, thermal fluctuations in global quantities are negligible and relatively simple relations of low-dimensional macroscopic variables are often sufficient to describe the whole system [148, 149]. As another consequence, the microcanonical ensemble (fixed energy) and the canonical ensemble (fixed temperature) are statistically equivalent, as the energy fluctuations in the canonical ensemble become negligible and the energy essentially has a unique value [150].

3.2.2 Latent Variables

Switching between representations of different resolutions is a hallmark of MS modeling. However, a similar concept is also well known in the ML community, where hidden and typically lower-dimensional representations are described by so called *latent variables*. ML models that relate latent with observable variables are referred to as *latent variable models* (LVM)[151, 152].

LVMs are motivated by the assumption that real-world data is generally structured, which implies that it can be described through a lower-dimensional latent distribution $\mathcal{Z}$ supported in $\mathbb{R}^d$. This assumption is known as the *manifold hypothesis*. Formally, it states that high-dimensional data $\mathbf{x} \in \mathbb{R}^D$ tends to lie on a low-dimensional manifold $\mathcal{M} \subset \mathbb{R}^D$ embedded in the high-dimensional ambient space

$\mathbb{R}^D$ [153]. Therefore, real-world data is assumed to have an intrinsic lower dimensionality $d < D$ and the observations $\mathbf{x}$ can be explained through some hidden variables $\mathbf{z} \in \mathbb{R}^d$. However, the actual dimension d of the latent space is typically unknown, as well as the mapping between latent $\mathbf{z}$ and ambient variables $\mathbf{x}$.

In practice, the concept of coarse-graining and latent variables ultimately follow the same philosophy: Both aim at reducing the complexity by discovering a lower-dimensional, simpler representation that still captures the essential features, while noise and redundant features are removed. However, coarse-grained representations of molecular systems are typically based on physical and chemical intuition, whereas dimensionality reduction based on LVMs deploys a learning scheme to discover the hidden and potentially lower-dimensional representation instead, i.e. a cost function is minimized. Recently, LVMs have been applied to the coarse-graining of molecular systems [126].

Similarly to the concept of backmapping, LVMs can be applied to the inverse task as well, i.e. to generate new instances of $\mathbf{x}' \sim \mathcal{X}$. In fact, this thesis deploys a LVM for the backmapping of molecular structures. For this purpose, the generative process is decomposed into two steps: (1) Draw a sample $z \sim \mathcal{Z}$ from the latent distribution with probability $p_{\mathcal{Z}}(z)$. (2) Introduce a ML model

$$g_{\Theta} : \mathbb{R}^d \to \mathbb{R}^D, \tag{3.11}$$

that transforms points from $\mathcal{Z}$ in order to resemble the (intractable) ambient distribution, i.e. $g(\mathcal{Z}) \approx \mathcal{X}$. Note that deriving g_{Θ} from first principles is infeasible for most real-world applications. Typically, a tractable distribution, such as a Gaussian distribution, is deployed as a latent distribution. However, the proposed $\mathcal{Z}$ can differ significantly from the actual manifold the data resides. As such, a linear transformation is not sufficient to deform $\mathcal{Z}$ in order to match $\mathcal{X}$ [154]. Consequently, highly nonlinear transformations are typically used, such as deep neural networks, which will be introduced in Sec. 3.3 [155, 156].

3.2.3 Dimensionality Reduction Algorithms

Reducing the dimensionality has become an important aspect of modern data analysis. Algorithms designed for this purpose are numerous and range from classical linear techniques, such as principle component analysis (PCA), to nonlinear ML approaches, such as LVMs. This section gives a brief overview of some important techniques used in this thesis for analyzing molecular data in order to assess the accuracy of reverse-mapped structures.

Principle Component Analysis

One of the most commonly used methods for dimensionality reduction is PCA. It performs a change of basis by projecting the data onto a linear combination of the original basis vectors called principle components. The new coordinate system is thereby chosen such that the axis align with the directions showing the highest variance in the data set. PCA is typically used for dimensionality reduction by discarding principle components with low variance, i.e. the new representation preserves most of the variance in the data.

Sketch-map

Despite its wide range of applications, linear dimensionality reduction techniques like PCA are typically insufficient to capture the global structure of data obtained from MD trajectories, since the accessible regions in phase space can have a complex, nonlinear structure with non-uniform dimensionality [157, 158, 159]. As such, nonlinear dimensionality reduction techniques are more promising candidates to analyze the phase space of molecular systems.

A successful approach introduced by Ceriotti *et al.* is called *Sketch-map* (SM) [160, 161]. Related to the curse of dimensionality, Ceriotti *et al.* hypothesize that energetically accessible regions in phase space are concentrated in basins and consequently, only a tiny fraction of phase space is occupied [160]. Further, small pairwise distances in phase space appear to follow a Gaussian distribution, which is expected for thermal fluctuations within a basin. On the contrary, large distances, which can be associated with structures that lie within different basins, are found to be uniformly distributed. Consequently, the focus has to be set on an intermediate scale σ capturing most of the valuable topological information of the phase space.

The SM approach is a nonlinear dimensionality reduction algorithm that aims at preserving the complex structure of the high-dimensional phase space. In particular, the method focuses on reproducing the relations between nearby basins, while the internal structure of basins and the relative positions of distant basins are ignored. To this end, distances in the high- and low-dimensional spaces are transformed by a sigmoid function. In particular, the following cost function is minimized to obtain projections $\mathbf{z}$ for N high-dimensional data points $\mathbf{x}$,

$$\mathcal{C}(\{\mathbf{z}_1, \dots, \mathbf{z}_N\}) = \sum_i^N \sum_j^N \Big(f(r_{ij}) - F(R_{ij}) \Big)^2, \tag{3.12}$$

where r_{ij} and R_{ij} correspond to the low-dimensional and high-dimensional distance of two points $\mathbf{z}_i$ and $\mathbf{z}_j$, as well as $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. F and f are sigmoid functions of the form

$$F(R) = 1 - (1 + (2^{A/B} - 1)(R/\sigma)^A)^{-B/A} \tag{3.13}$$

$$f(r) = 1 - (1 + (2^{a/b} - 1)(r/\sigma)^a)^{-b/a}, \tag{3.14}$$

where σ, A, B, a and b are the parameters of the model. The value of σ is the most important parameter, as it defines the characteristic length scale of the lower-dimensional embedding. In practice, σ has to lie within the range of distances characteristic of Gaussian fluctuations and the range dominated by the high dimensionality of the system. The sigmoid function transforms distances such that distances far below or far apart from σ are mapped close to zero or unity, respectively. As such, distances in the vicinity of σ are highlighted. The minimization of Eq. 3.12 scales quadratically with the number of data points. To reduce the computational effort for projections of large data sets, Eq. 3.12 is used to obtain low-dimensional positions for a small number of landmark frames. Afterwards, the landmarks can be used to project any other data point $\mathbf{x}$ by minimizing

$$\mathcal{C}(\mathbf{z}) = \sum_{i}^{N_L} \Big(f(|\mathbf{z} - \mathbf{z}_i|) - F(|\mathbf{x} - \mathbf{x}_i|) \Big)^2, \tag{3.15}$$

where $\mathbf{z}_i$ is the projection of a landmark $\mathbf{x}_i$ and N_L is the number of landmarks.

Time-lagged Independent Components Analysis

Time-lagged independent component analysis (TICA) is a linear transformation algorithm that is widely used for dimensionality reduction of MD trajectories [162, 163, 164, 165]. While PCA finds coordinates of maximal variance, TICA aims at a mapping that maximizes the autocorrelation at the given lag time, i.e. it identifies the slow degrees of freedom. As such, TICA explores a subspace of good reaction coordinates and is well suited to prepare high-dimensional input data for Markov model construction.

Consider a mean-free trajectory of configurations $\mathbf{x}(t) \in \mathbb{R}^D$, where t is an integer denoting the time step. The covariance matrix $\mathbf{C}(\tau)$ of the data is obtained as

$$c_{ij}(\tau) = \langle x_i(t) x_j(t+\tau) \rangle_t, \tag{3.16}$$

where τ is the lag time. In general, $\mathbf{C}$ has to be symmetrized for algebraic reason. TICA proceeds by solving the generalized eigenvalue problem

$$\mathbf{C}(\tau)\mathbf{U} = \mathbf{C}(0)\mathbf{U}\mathbf{\Lambda}, \tag{3.17}$$

where $\mathbf{U}$ is a eigenvector matrix composed of the independent components (ICs) and $\mathbf{\Lambda}$ is a diagonal eigenvalue matrix. Deploying the eigenvector-matrix $\mathbf{U}$, projections into the latent space are given by

$$\mathbf{z}^T(t) = \mathbf{x}^T(t)\mathbf{U}. \tag{3.18}$$

Similar to PCA, the eigenvector matrix $\mathbf{U}$ can be truncated to establish dimensionality reduction that keeps the slowest modes.

3.3 Deep Learning

Deep Learning is a part of ML that is based on *artificial neural networks* (ANNs). ANNs are inspired by nervous systems, such as our human brain. A nervous system processes information and is capable to perform extremely complex tasks: It coordinates incoming signals, such as information about the environment captured by sensory cells, and creates actions accordingly. Their ability to organize themselves and learn from examples distinguishes them from conventional computers and has motivated researchers to develop artificial models of its biological counterpart [166, 167, 168]. Nowadays, ANNs have won several state of the art ML contests and are used in many applications [169]. In this work, ANNs are the main tool for the reverse-mapping of molecular configurations.

3.3.1 General Concept

Nervous systems are built up from a large number of interconnected cells, called *neurons*. The human brain, as an example, consists of $\sim 10^{11}$ neurons [170]. While a single neuron is already a complex processing unit, the power of nervous systems arises from the interplay of a vast number of neurons composing the network.

The main task of a neuron is to receive, process and transmit signals. In a simplified picture, a biological neuron is equipped with *dendrites* (receiver), a *cell body* (processor) and an *axon* (transmitter) [170]. Dendrites are thin fibers connected via *synapses* with the axons of thousands of other neurons. Synapses are crucial for the flow of information inside the network, as they weight incoming signals captured by the dendrites: Depending on the synapse, the incoming signal can either increase or decrease the electrical potential of the cell [170]. If a specific *threshold potential* is reached, the axon will fire a signal to all the dendrites it is connected to.

An artificial neuron is a simplified version of its biological counterpart: An *input tensor* $\mathbf{x}$ is weighted by a *weight tensor* $\mathbf{w}$ and the result is accumulated. Afterwards, a *threshold value* ψ is subtracted and a nonlinear function, called *activation function*, $a()$ is applied to derive the output y,

$$y = a\Big(\sum_i x_i w_i - \psi\Big). \tag{3.19}$$

In this example, $\mathbf{x}$ and $\mathbf{w}$ are vectors but it is straightforward to extend the concept to tensors of higher rank [171].

3.3.2 Multiple Levels of Abstraction

To explain the mechanism of information processing in a biological neural network, visual object recognition can serve as an example: The retina encodes visual stimuli into electrical signals that are transmitted to the visual cortex. Here, the incoming signal will cause a subset of neurons to respond yielding a response vector [172]. The response vector for a given object is not constant but varies under identity preserving transformations, such as shifts in position, rotations or changing illumination. Therefore, a given object has to be linked to a set of response vectors that span a manifold in the high-dimensional space of all possible response vectors [172]. At early stages of processing, the object identity manifolds for different objects might be tangled. As such, it becomes impossible to introduce an accurate decision boundary for object recognition. However, the special structure of the visual cortex enables to untangle the object manifolds. In particular, neurons are grouped into subsequent layers and the further the signals are processed the more flattened and separated the manifolds become [172].

Deep Neural Networks

The same idea is applied in modern deep neural networks (DNNs): Multiple layers are arranged subsequently and each layer transforms its input into a more abstract and composite representation. While the first layer learns simple features, such as the positions of edges, subsequent layers learn more complex features composed of the preceding ones. As such, DNNs are computational models that are similar in spirit to the multiscale modeling approach.

The layers arranged between the input layer and the output layer are referred to as *hidden layers*. In the most simple case of a *feedforward neural network* (FF-NN), information only flows forward in the network, i.e. from the input layer through the hidden layers to the output layer. In particular, a simple FF-NN f consisting of L layer can be written as

$$f(\mathbf{x}) = \mathbf{u}^{(L)}(...\mathbf{u}^{(1)}(\mathbf{u}^{(0)}(\mathbf{x}))), \tag{3.20}$$

where $\mathbf{u}^{(l)}$ denotes the nodes in the l'th layer of the network.

Recurrent Neural Networks

Different to FF-NNs, *recurrent neural networks* (RNNs) can have a more complex architecture. RNNs enable cyclic connections between layers, such that a layer can receive information from subsequent layers as well in order to create feedback loops [173]. In particular, a RNN can be described as a recursive process where a recurring function is called in each iteration. This is usually necessary for sequential data, like text or video, where the current state has a dependence on past states. In this thesis, a RNN is used to recursively reconstruct molecular structures.

Unrolling the recurrent process yields a nested function that is similar to a FF-NN. For example, consider a network f that reuses its previous output in addition to its current input. The unrolled network F for an input sequence $(\mathbf{x}_1, \mathbf{x}_2, .., \mathbf{x}_n)$ can be written as

$$F(\mathbf{x}_1, \mathbf{x}_2, .., \mathbf{x}_n) = f(\mathbf{x}_n, ..f(\mathbf{x}_2, f(\mathbf{x}_1, 0))..). \tag{3.21}$$

Note, that Eq. 3.21 implies that multiple states of the network have to be stored simultaneously in memory in order to compute the gradients for training (see Sec. 3.3.4) making the approach computationally demanding.

Universal Function Approximator

DNNs are universal function approximators. In particular, the classical formulation of the *universal function approximation theorem* states that any continuous function h on a compact set K can be approximated by a FF-NN f with just one hidden layer within arbitrary accuracy $|f(x) - h(x)| < \epsilon$, where $\epsilon > 0$ and $x \in K$ [174]. Importantly, the *width* of the hidden layer, i.e. the number of neurons, needs to be unbound in this formulation of the theorem. A dual formulation states that the theorem holds true for bounded width but arbitrary *depth*, i.e. number of layers, as well [175].

Generalization Capability

Despite being universal function approximators, DNNs also exhibit good generalization behavior. This is surprising, since DNNs are typically over-parameterized, i.e. the number of parameters of the network is significantly larger than the number of training examples. As such, the network could memorize the training samples, i.e. overfit the training data, instead of learning the underlying rules that generate the data. As explained in Sec. 3.1.2, regularization is typically used to prevent overfitting by limiting the complexity of the model. This can be achieved by explicit methods, such as penalty terms in the cost function or parameter decay. However, studies have shown empirically that explicit regularization is neither a sufficient nor a necessary condition for the generalization capability of DNNs [176]. Moreover, implicit regularization techniques, such as stochastic optimization procedures or early stopping of the training, can improve the generalization capability of DNNs, but are also not indispensable [177]. It can be hypothesized that the generalization capability of DNNs is inherently linked to their architecture [178]. In conclusion, a more comprehensive theory that explains why DNNs generalize well in practice is an open area of research [179, 180, 181].

3.3.3 Feature Extraction

An ANN is a composition of layers, where each layer generates features based on the representation produced by the previous layer. Unlike traditional approaches, relevant features are learned from the data. As such, ANNs can learn a suitable representation of the data for a given task without relying on handcrafted features. The feature extraction at each level of an ANN can be achieved in various different ways. In the following, an overview of the two most commonly used layer architectures is given.

Dense Layer

A *dense layer* or *fully-connected layer* is the simplest and most common layer of an ANN. Each neuron in a dense layer receives signals $\mathbf{x}$ from all the neurons of the preceding layer. The output $\mathbf{y}$ of a dense layer can be expressed as

$$\mathbf{y} = a(\mathbf{A}\mathbf{x}), \tag{3.22}$$

where $\mathbf{A}$ is a matrix containing the weights and thresholds. The full-connectivity of each neuron makes it capable to detect global pattern in the data. However, dense layer are impractical for high-dimensional inputs, as $\mathbf{A}$ grows with the size of $\mathbf{x}$.

Convolutional Layer

Convolutional neural networks (CNNs) have dramatically improved the state-of-the-art in computer vision and made it possible to synthesize photorealistic images of complex objects, such as human faces or animals [6, 50, 51, 52]. Therefore, CNNs are the main tool in this thesis to synthesize molecular structures.

Convolutional layer incorporate the idea of local connectivity and translational-equivariance. The idea of local connectivity is inspired by the visual cortex: Neurons are only locally connected to neurons in a restricted area of the previous layer, known as the *receptive field* [170]. The receptive field of a neuron becomes bigger the higher it is placed in the hierarchy of the network. Therefore, this architecture is perfectly suited to learn hierarchical pattern in the data.

A convolutional layer consists of a bank of parameterized *filters* that are sliced over the input. In the following, the two dimensional case is considered, i.e. images $\mathbf{X}_j \in \mathbb{R}^{N_x \times N_y}$, where N_x is the width and N_y is the height of the image. The subscript $j \in \{0, .., N_c\}$ is the index for the N_c *feature channels*. The different feature channels provide different views on the data, such as the different color channels of a red-green-blue (RGB) image. At each position, the discrete convolution of the filter with the segment of the image it overlaps is computed and stored into a so called *feature map*. Note that the parameters of each filter are shared between all neurons of the layer, which enforces equivariance with respect to translations of the learned patterns. The bank of filters connecting the jth feature channel of the input with the ith feature channel of the output is denoted with $\mathbf{K}_{i,j} \in \mathbb{R}^{m_x \times m_y}$, where m_x is the width, m_y is the height of the filters. The ith feature map $\mathbf{Y}_i \in \mathbb{R}^{n_x \times n_y}$ of the output can be computed as

$$\mathbf{Y}_i = a(\mathbf{\Phi} + \sum_{j}^{N_c} \mathbf{K}_{i,j} * \mathbf{X}_j), \tag{3.23}$$

where $\mathbf{\Phi}$ is a bias matrix. The size of the output (n_x, n_y) can be derived as

$$(n_x, n_y) = (N_x - m_x + 1, N_y - m_y + 1). \tag{3.24}$$

Additionally, the size of the output can be controlled by *zero-padding* and *slicing*:

- Zero-padding: The size of the input tensor can be artificially extended by adding zeros at the border or between input units. P denotes the number of zeros concatenated at each side.
- Strides: While the filter is sliced over the input, the step size S, called *stride*, for the translation can be greater than one effectively reducing the output size, or smaller than one effectively increasing the output size.

In summary, the output size can be computed as

$$(n_x, n_y) = \Big(\frac{N_x - m_x + 2P}{S} + 1, \frac{N_y - m_y + 2P}{S} + 1\Big). \tag{3.25}$$

Depending on the choice of zero-padding and strides, the size of the output can either decrease (downsampling) or increase (upsampling) compared to the size of the input. The latter is often referred to as *transposed* (or *fractionally-strided*) convolution. It is typically used in a decoder architecture to learn the upsampling transformation. Note that it is possible to express a convolutional layer as a fully-connected layer as well. However, this is not done in practice, as this involves many unnecessary multiplications with zero. In general, a convolutional layer requires less parameters compared to a dense layer, because of the weight-sharing of the filters and the independence of the size of the filters from the size of the input.

3.3.4 Training

Training of a NN $f_{\mathbf{\Theta}}$ refers to tuning the weights $\mathbf{\Theta}$ in order to minimize a cost function. To achieve this, an efficient optimization algorithm is required.

Cost Function

The cost function $\mathcal{C}$ maps the output of the network $f_{\mathbf{\Theta}}(\mathbf{x})$ to a real number representing the error. While more details about the cost function used in this thesis will be revealed in in Sec. 3.4, two requirements it has to fulfill shall be noted to in order to explain the training procedure of ANNs: (1) It has be differentiable in order to apply gradient-based optimization algorithms. (2) It has be written as an average

$$\overline{\mathcal{C}}_T = \frac{1}{n} \sum_{(\mathbf{x},\mathbf{y}) \in T} \mathcal{C}(f_\Theta(\mathbf{x}), \mathbf{y}) \tag{3.26}$$

over cost functions $\mathcal{C}(f_{\mathbf{\Theta}}(\mathbf{x}), \mathbf{y})$ for individual instances of the training batch $T = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_n, \mathbf{y}_n)\}$. This is crucial in order to generalize the gradient of the error computed for a single example to the overall error of the training batch (see Stochastic Gradient Descent below).

Backpropagation

In order to adjust the weights $\mathbf{\Theta}$ of the network, gradient methods are deployed. Typically, gradient descent is applied for single input-output pairs $(\mathbf{x}, \mathbf{y})$ in the weight space,

$$\mathbf{\Theta}_{t+1} = \mathbf{\Theta}_t + \eta \frac{\partial \mathcal{C}(f_{\mathbf{\Theta}_t}(\mathbf{x}), \mathbf{y})}{\partial \mathbf{\Theta}_t} \tag{3.27}$$

where η is the learning rate and t is an integer denoting the optimization step.

A naive, direct computation of the gradients with respect to each weight individually is computationally expensive and not feasible for DNNs. To circumvent these limitations, *backpropagation* (BP) was invented to efficiently compute the gradients for DNNs [182]. It is based on the chain rule and benefits from the nested structure of a NN, which enables to compute the gradients layer by layer: Starting from the last layer, it iterates backward through the network, whereby avoiding duplicate and unnecessary intermediate calculations.

Consider a feed-forward NN with L layers. In the following, each layer is treated as a fully connected layer for simplicity. A single node $u_i^{(l)}$ in layer l can be written as

$$u_j^{(l)} = a(\underbrace{\sum_i \Theta_{i,j}^{(l)} u_i^{(l-1)}}_{z_j^{(l)}}), \tag{3.28}$$

where $\mathbf{\Theta}^{(l)}$ is the weight matrix for layer l and $a()$ is the activation function.

The chain rule has to be applied to derive the gradients for the weights due to the nested structure of the NN. Conveniently, the BP algorithm introduces a recursive notation to derive the gradients,

$$\frac{\partial \mathcal{C}(f_{\mathbf{\Theta}}(\mathbf{x}), \mathbf{y})}{\partial \Theta_{i,j}^{(l)}} = \delta_j^{(l)} u_i^{(l-1)}, \tag{3.29}$$

where $\delta_j^{(l)}$ is referred to as the *delta-error* or *error at the level l*. It is computed as

$$\delta_j^{(l)} = \begin{cases} \mathcal{C}' a'(z_j^{(l)}), & \text{for } l = L \\ a'(z_j^{(l)}) \sum_i \Theta_{i,j}^{(l)} \delta_j^{(l+1)}, & \text{for } l < L \end{cases}, \tag{3.30}$$

where $\mathcal{C}'$ and a' are the derivatives of the cost function $\mathcal{C}$ and the activation function a, respectively. The recursive notation for the delta-error enables to compute the required gradients from back to front by reusing the delta-error from subsequent layers.

Stochastic Optimization

Finding the global minima in a high-dimensional energy landscape is challenging. In particular, naive deterministic optimization algorithms, such as gradient descent (Eq. 3.27), are prone to get stuck in local minima. To this end, stochastic optimization algorithms are typically deployed that permit less optimal local decisions in order increase the probability of eventually deriving the global minima.

A widely used stochastic optimization algorithm is *stochastic gradient descent* (SGD): The size of a training set is typically large and therefore, the gradient in Eq. 3.27 can not be computed over the whole set in each optimization step. Therefore, the training set is usually shuffled and split into mini-batches $T = \{(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_n, \mathbf{y}_n)\}$ of size n, where n is treated as a hyper parameter. Since the gradient is not computed exactly but only for a part of the data, the optimization algorithm contains a stochastic element. However, the gradient estimation can vary significantly for different training batches in SGD. To this end, SGD can be augmented by a momentum term

$$\mathbf{\Theta}_{t+1} = \mathbf{\Theta}_t + \eta \mathbf{v}_t, \tag{3.31}$$

$$\mathbf{v}_t = \beta \mathbf{v}_{t-1} + (1 - \beta) \frac{\partial \mathcal{C}(f_{\mathbf{\Theta}_t}(\mathbf{x}), \mathbf{y})}{\partial \mathbf{\Theta}_t}, \tag{3.32}$$

where β is the momentum parameter. Incorporating momentum smooths the gradient and improves consistency between optimization steps.

3.4 Generative Modeling

While DNNs will built the core of the reverse-mapping scheme developed in this thesis, an adequate cost function to train the model is required. In particular, the model has to be trained such that it can synthesize further samples from the fine-grained distribution of a molecular system. To this end, concepts of *generative modeling* are applied to learn this distribution from training data.

Generative models can be distinguished from discriminative models in terms of their purpose: While the former aims at learning the dependencies of all the variables in a system, the goal of the latter is to learn decision boundaries. Consider a set of labeled data $\mathcal{D} = \{(\mathbf{x}, \mathbf{y})\}$ drawn from a distribution $\mathcal{X}$ with joint probability $p_{\mathcal{X}}(\mathbf{x}, \mathbf{y})$. The goal of a discriminative model is to learn the conditional probability $p_{\mathcal{X}}(\mathbf{y}|\mathbf{x})$ of the class labels $\mathbf{y}$ given the observation $\mathbf{x}$. To this end, a discriminative model does not necessarily have to learn about the dependencies of all the variables in the system, as it only has to focus on the variables that are important to label the observations. On the other hand, generative approaches model the joint probability $p_{\mathcal{X}}(\mathbf{x}, \mathbf{y})$ (or simply $p_{\mathcal{X}}(\mathbf{x})$ if no labels are given). As such, the underlying task for generative models is more complex compared to the discriminative task, as the generative model has to be more informative.

Since only generative models are capable of generating new instances of the underlying distribution, the remainder of this chapter focuses on the generative approach. More specifically, the following discussion is restricted to *Deep Generative Models* (DGMs), i.e. generative models based on DNNs.

3.4.1 Maximum Likelihood and the Relative Entropy

A DGM provides an estimate for the probability $p_{\Theta}(\mathbf{x})$ of an observation $\mathbf{x}$. The general goal is to approximate the true probability distribution, i.e. to find the optimal parameters Θ^* such that $p_{\Theta^*}(\mathbf{x}) \approx p_{\mathcal{X}}(\mathbf{x})$. A major route to train a DGM is the frequentist approach of maximizing the data likelihood. The underlying idea of this approach is to find a model that best explains the observed data. In particular, the likelihood $\mathcal{L}$ for the data under the parametric model can be written as

$$\mathcal{L} = \prod_{i}^{N} p_{\Theta}(x), \tag{3.33}$$

where N is the number of examples in the training data $\mathcal{D}$. Instead of maximizing the likelihood $\mathcal{L}$ directly, it is common practice to minimize the negative logarithm of the likelihood to avoid numerical issues

$$\Theta^* = \underset{\Theta}{\text{argmax}}\ \mathcal{L} = \underset{\Theta}{\text{argmin}}\ -\sum_{i}^{N} log(p_{\Theta}(\mathbf{x})). \tag{3.34}$$

If N is large, the maximum likelihood approach is equivalent to minimizing the *cross-entropy*

$$H(p_{\mathcal{X}}(\mathbf{x}), p_{\Theta}(\mathbf{x})) = \mathbb{E}_{x\sim p_{\mathcal{X}}(\mathbf{x})}\Big[log\Big(\frac{1}{p_{\Theta}(\mathbf{x})}\Big)\Big], \tag{3.35}$$

as well as the *relative entropy*, which is also known as the *Kullback-Leibler divergence*

$$D(p_{\mathcal{X}}(\mathbf{x})||p_{\Theta}(\mathbf{x})) = \mathbb{E}_{x\sim p_{\mathcal{X}}(\mathbf{x})}\Big[log\Big(\frac{p_{\mathcal{X}}(\mathbf{x})}{p_{\Theta}(\mathbf{x})}\Big)\Big] \tag{3.36}$$

as the law of large numbers states

$$\lim_{N\to\infty}\ -\frac{1}{N}\sum_{i}^{N} log(p_{\Theta}(\mathbf{x})) =\ \mathbb{E}_{x\sim p_{\mathcal{X}}(\mathbf{x})}[-log(p_{\Theta}(\mathbf{x}))], \tag{3.37}$$

and the scaling factor $\frac{1}{N}$ is irrelevant, as it does not affect the $\underset{\Theta}{\text{argmin}}$ operation. Both, cross-entropy and relative entropy, reach a minima when the two distributions match, i.e. $p_{\Theta}(\mathbf{x}) = p_{\mathcal{X}}(\mathbf{x})$, making the negative logarithm of the likelihood a well suited cost function for generative models. However, how can the likelihood function be assessed?

3.4.2 Review of Explicit Generative Models

Generative models are further distinguished between those that express the model probability distribution $p_{\Theta}(\mathbf{x})$ *explicitly* through some functional form and those that define it *implicitly* through a sampler. The former approach has the benefit that maximizing the likelihood $\mathcal{L}$ is straightforward, as the probability distribution can be assessed directly. However, explicit models require to assume some functional form for the probability $p_{\Theta}(\mathbf{x})$, which often becomes a bottleneck for the expressive power of the model. Therefore, the design of an explicit model is often a tradeoff between the complexity and tractability of the model. In particular, the various variants of explicit DGMs fall into two categories: Models that carefully construct a tractable functional form for the likelihood $\mathcal{L}$, such as autoregressive models and normalizing flow models, and models that use a tractable approximation, such as the variational auto encoder and the Boltzmann machine [183]. On the other hand, implicit models do not require direct access of the likelihood function but define a stochastic procedure to generate new samples. As such, the implicit approach is perfectly suited for the overall goal of thesis, i.e. to generate new fine-grained configurations. In particular, the generative adversarial approach is used, which is the most prominent member of implicit generative models.

In the following, important explicit generative models are reviewed briefly. Afterwards, the generative adversarial approach is explained in detail.

Autoregressive Modeling

The autoregressive approach decomposes a complex probability distribution into simpler conditional probability distributions [184, 185, 186]. To this end, the chain rule for probabilities is applied to rewrite the joint probability $p(\mathbf{x})$ of an n-dimensional vector $\mathbf{x}$ into a product of conditional probabilities that are easy to access

$$p_{\Theta}(\mathbf{x}) = \prod_{i}^{n} p_{\Theta}(x_i|x_1, x_2, .., x_{i-1}). \tag{3.38}$$

Splitting the problem into conditional probabilities often allows for a tractable explicit model. The drawback of such autoregressive models is that they can only generate one entry at a time prohibiting parallel computation. However, this approach is well suited for data that is sequential in nature, such as human speech. The autoregressive approach is also important in this thesis, as it will be used to factorize the joint probability of a molecular configuration in terms of atomic contributions to reduce the complexity of the reverse-mapping task.

Normalizing Flow

Normalizing Flow (NF) models are LVMs that transform a simple prior distribution into a more complex distribution using invertible and differentiable mappings [187, 188, 189]. NF models are based on the change of variables formula for invertible transformations g that map from the data distribution $\mathcal{X}$ to the latent distribution $\mathcal{Z}$. Given a transformation

$$g : \mathrm{R}^D \to \mathrm{R}^D \tag{3.39}$$

that is invertible and both g and g^{-1} are continuously differentiable, as well as orientation-preserving, i.e. $\nabla g > 0$, then the change of variables formula can be used to express the likelihood of a data point $x \sim \mathcal{X}$ in terms of another, potentially simpler, density function in the latent space

$$p_{\mathcal{X}}(\mathbf{x}) = p_{\mathcal{Z}}(g^{-1}(\mathbf{x}))\left|\det\left(\frac{\partial g^{-1}(\mathbf{x})}{\partial \mathbf{x}}\right)\right|. \tag{3.40}$$

NF models have the advantage that they allow a direct optimization and evaluation of the likelihood. However, major drawbacks of NF models are the required restrictions on the transformation g that are difficult to fulfill in practice and limit their applicability [140]. The most severe restriction is the equality of the dimensions of the latent and the data space.

Variational Autoencoder

A variational autoencoder (VAE) is a LVM consisting of two parts [190, 191, 192]: An encoder $e_\Psi(\mathbf{z}|\mathbf{x})$ compresses a given input $\mathbf{x}$ into a constraint distribution in the latent space $\mathbb{R}^d$ and a decoder $p_\Theta(\mathbf{x}|\mathbf{z})$ reconstructs the distribution in the ambient space $\mathbb{R}^D$. Typically, $e_\Psi(\mathbf{z}|\mathbf{x})$ and $p_\Theta(\mathbf{x}|\mathbf{z})$ are represented as a family of parameterized distributions, such as multivariate Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_\Psi, \boldsymbol{\Sigma}_\Psi)$ and $\mathcal{N}(\boldsymbol{\mu}_\Theta, \boldsymbol{\Sigma}_\Theta)$. In particular, the means $\boldsymbol{\mu}$ and variances $\boldsymbol{\Sigma}$ of the distributions are learned by DNNs with weights Ψ and Θ, respectively. Importantly, unlike NF models, the dimensions of latent and ambient space do not have to match. In general, the dimension of the latent space d is chosen much smaller than the dimension of the ambient space D. As a consequence, the mapping between the spaces is not invertible and the likelihood can not be computed directly.

Typically, a standard normal distribution is deployed as a prior distribution $p_{\mathcal{Z}}(\mathbf{z})$ for the latent variables. Using Bayes's rule, the likelihood for a data point $\mathbf{x}$ can be written as

$$p_\Theta(\mathbf{x}) = \frac{p_\Theta(\mathbf{x}|\mathbf{z})p_{\mathcal{Z}}(\mathbf{z})}{p_\Theta(\mathbf{z}|\mathbf{x})}. \tag{3.41}$$

While the posterior distribution $p_\Theta(\mathbf{z}|\mathbf{x})$ is intractable in most cases, the VAE approach approximates it deploying the encoder $e_\Psi(\mathbf{z}|\mathbf{x})$,

$$e_\Psi(\mathbf{z}|\mathbf{x}) \approx p_\Theta(\mathbf{z}|\mathbf{x}). \tag{3.42}$$

Using Jensen's inequality, a variational lower bound, also called evidence lower bound (ELBO), can be derived to train both, the encoder e_Ψ and the decoder $p_\Theta(z|x)$:

$$
\begin{aligned}
log(p_{\Theta}(\mathbf{x})) &\geq \mathbb{E}_{e_{\Psi}(\mathbf{z}|\mathbf{x})}\Big[log\big(\frac{p_{\Theta}(\mathbf{x},\mathbf{z})}{e_{\Psi}(\mathbf{z}|\mathbf{x})}\big)\Big] && (3.43)\\
&= \underbrace{\mathbb{E}_{e_{\Psi}(\mathbf{z}|\mathbf{x})}\Big[log\big(p_{\Theta}(\mathbf{x}|\mathbf{z})\big)\Big]}_{\text{reconstruction}} - \underbrace{\mathrm{KL}(e_{\Psi}(\mathbf{z}|\mathbf{x})||p_{\mathcal{Z}}(\mathbf{z}))}_{\text{regularization}} && (3.44)
\end{aligned}
$$

The negative ELBO is then minimized. The first term reduces the approximation error in ambient space (reconstruction error) that arises due to the restriction for the dimensionality of the latent space and the approximation error for the posterior. The second term acts as a regularizer that biases the approximate posterior towards the prior distribution $p_{\mathcal{Z}}(\mathbf{z})$.

The main drawback of VAEs is a potentially large gap between the ELBO used for optimization and the actual likelihood resulting in a model that differs significantly from the true distribution [140]. Empirically, vanilla VAEs have a tendency to ignore some of the latent variables and/or produce blurred samples in ambient space [193].

Markov Chain approximation

Some generative models generate samples using a Markov chain technique: A sample $\mathbf{x}$ is repeatedly updated according to some transition operator $\mathbf{x}' \sim q(\mathbf{x}'|\mathbf{x})$. An example is the Boltzmann machine (BM), which is a LVM that consists of binary units u_i connected with each other [194, 133, 195, 196]. The weights Θ and thresholds Φ of the network are essentially the parameters of an energy function $E(\Theta, \Phi)$. In particular, $E(\Theta, \Phi)$ describes a spin-glass model with an external field

$$
E(\Theta, \Phi) = -\Big(\sum_{i<j} \Theta_{i,j} u_i u_j + \sum_i \Phi_i u_i\Big). \tag{3.45}
$$

The energy function can be used to define a probability distribution over the states of the units. During training, the weights are updated such that the likelihood for the given training data is maximized. To sample a new state, the units are repeatedly updated stochastically until an equilibrium state is reached.

The convergence of such Markov models might be very slow in practice and difficult to detect [183]. BMs and its variants are barely used nowadays, because they do not scale well to high-dimensional data.

3.4.3 The Generative Adversarial Network: An Implicit Generative Model

Generative adversarial networks (GANs) were introduced by Ian Goodfellow *et al.* in 2014 [197]. They have become one of the most successful implicit generative models known in the ML community [50, 51, 198, 199]. Their ability to generate photorealistic images of complex objects have motivated their usage in this thesis as the main training procedure to generate high fidelity molecular structures.

A GAN is a LVM, but unlike VAEs or NF models, they do not infer the distribution of latent variables underlie the samples. Instead, they learn a transformation from a given prior distribution $\mathcal{Z}$ to an ambient distribution $\mathcal{X}$, which is for example the distribution of molecular configurations. At its core, a GAN consists of two competing models trained in a game: A generator g_Θ maps samples $\mathbf{z} \in \mathbb{R}^d$ from a latent distribution $\mathcal{Z}$ into the ambient space $\mathbb{R}^D$. A second model, the discriminator c_Ψ, has to distinguish between synthetic samples $g_\Theta(\mathbf{z})$ from the generator and real samples $\mathbf{x}$ from the training set $\mathcal{D} = \{\mathbf{x}\}$, where $\mathbf{x}$ are drawn from $\mathcal{X}$. As such, the discriminator c_Ψ acts as a distance measure in ambient space for the real distribution $\mathcal{X}$ and the distribution of synthetic samples $g_\Theta(\mathcal{Z})$. While the discriminator c_Ψ is trained as a classifier in a supervised manner, the generator g_Θ is trained to deceive the discriminator c_Ψ. As a consequence, the generator g_Θ is indirectly pushed towards minimizing the difference between $\mathcal{X}$ and $g_\Theta(\mathcal{Z})$. The whole training process of a GAN is considered a likelihood-free method as neither the likelihood of the model $p_\Theta(\mathbf{x})$ itself nor a lower bound of it is used explicitly.

Both, the generator g_Θ and the discriminator c_Ψ are typically implemented as DNNs with weights Θ and Ψ, respectively. The generator is an inverse LVM (see Sec. 3.2.2)

$$g_\Theta : \mathbb{R}^d \rightarrow \mathbb{R}^D, \tag{3.46}$$

that maps latent samples $\mathbf{z} \in \mathbb{R}^d$ into the ambient space $\mathbb{R}^D$. The prior distribution $\mathcal{Z}$ is typically defined as a high-dimensional Gaussian distribution or uniform distribution over a hypercube. Intuitively, the latent samples $\mathbf{z}$ provide a source of randomness for the model.

Vanilla Approach: Discriminator as Binary Classifier

In the seminal work of Ian Goodfellow *et al.*, the GAN training is set up as a binary classification problem, where the discriminator c_Ψ is a function

$$c_\Psi : \mathbb{R}^D \rightarrow [0, 1], \tag{3.47}$$

aiming to predict the probability whether a given sample is drawn from the distribution $\mathcal{X}$ or from the generator $g_\Theta(\mathcal{Z})$ [197]. As such, an optimal discriminator c_{Ψ^*} is supposed to predict $c_{\Psi^*}(\mathbf{x}) \approx 1$ and $c_{\Psi^*}(g(\mathbf{z})) \approx 0$.

The natural choice for the loss function for a binary-classification problem is the cross entropy. Therefore, the original cost function for the GAN $\mathcal{C}(g_\Theta, c_\Psi)$ is defined as [197]

$$\mathcal{C}(g_\Theta, c_\Psi) = \mathbb{E}_{\mathbf{x}\sim\mathcal{X}}\Big[log(c_\Psi(\mathbf{x}))\Big] + \mathbb{E}_{\mathbf{z}\sim\mathcal{Z}}\Big[log\Big(1 - c_\Psi(g_\Theta(\mathbf{z}))\Big)\Big]. \tag{3.48}$$

As a result, the GAN training becomes a mini-max game, where the discriminator aims at maximizing $\mathcal{C}\left(g_\Theta, c_\Psi\right)$, while the generator tries to minimize $\mathcal{C}\left(g_\Theta, c_\Psi\right)$ [197]:

$$\Psi^* = \arg\max_{\Psi} \mathcal{C}\left(g_\Theta, c_\Psi\right) \quad \text{and} \quad \Theta^* = \arg\min_{\Theta} \mathcal{C}\left(g_\Theta, c_\Psi\right) \tag{3.49}$$

This refers to a zero-sum game, where the gain of one player is the loss of the other. The training of a GAN converges when a saddle point (Ψ^*, Θ^*) is reached, which is also called *Nash equilibrium* in game theory: For both networks, the loss can not be optimized any further given the weights of the other.

Using the loss defined in 3.48, it can be shown that the optimal discriminator $c_{\Psi^*,\Theta}$ for a fixed generator g_Θ is given by [197]

$$c_{\Psi^*,\Theta}(\mathbf{x}) = \frac{p_{\mathcal{X}}(\mathbf{x})}{p_{\mathcal{X}}(\mathbf{x}) + p_\Theta(\mathbf{x})}. \tag{3.50}$$

Plugging Eq. 3.50 into Eq. 3.48 yields the cost function for the generator given an optimal discriminator,

$$\mathcal{C}(g_\Theta, c_{\Psi^*}) = 2JS(p_{\mathcal{X}}||p_\Theta) - 2\log(2), \tag{3.51}$$

where JS is the Jenson-Shannon divergence

$$JS(p_{\mathcal{X}}||p_\Theta) = \frac{1}{2}D\left(p_{\mathcal{X}}\middle|\middle|\frac{p_{\mathcal{X}} + p_\Theta}{2}\right) + \frac{1}{2}D\left(p_\Theta\middle|\middle|\frac{p_{\mathcal{X}} + p_\Theta}{2}\right). \tag{3.52}$$

As such, an optimal discriminator yields a cost function for the generator that minimizes the Jensen-Shannon divergence JS, which is a symmetrized variant of the relative entropy defined in Eq. 3.36.

Challenges of the GAN Approach

While the above analysis is instructive and motivates the usage of the GAN approach, the theoretical analysis does not hold in practice for several reasons: (1) The minimax game is tackled iteratively with an alternating approach, where the discriminator c_Ψ is trained for k steps in order to reach optimality followed by a single training step for g_Θ. Typically, a rather small number of training steps k for the discriminator is chosen in order to maintain a feasible optimization. Therefore, the assumption of an optimal discriminator does not apply and the convexity of the cost function is not guaranteed [197]. (2) Convergence to an equilibrium, where $p_\Theta(\mathbf{x}) = p_{\mathcal{X}}(\mathbf{x})$, is hindered by the limited capacity of the generator and the discriminator: A direct optimization in function space would be required to guarantee convergence. However, both models are represented as DNNs and optimization takes place in the finite parameter space [197, 200]. (3) Optimizing the generator with the cross entropy defined in Eq. 3.48 does not perform well in practice, as the cost can easily saturate [197, 200]: If the discriminator can reject samples produced by the generator with high confidence, the generator's gradients vanish making it impossible to improve. This limitation is especially severe in the beginning of the training, when generated samples are clearly different from the training examples. To circumvent this limitation, a heuristic loss for the generator

$$\mathcal{C}(g_\Theta) = \mathbb{E}_{\mathbf{z}\sim\mathcal{Z}}\Big[-log\Big(c_\Psi(g_\Theta(\mathbf{z}))\Big)\Big] \tag{3.53}$$

is typically used that provides strong gradients.

A more general issue of the GAN approach is mode collapse, which refers to a lack of diversity [201, 50, 202, 203]. If the generator has learned to generate a plausible output, it might overemphasize that specific output and put a overwhelmingly high statistical weight on it. Consequently, GANs tend to generate from very few modes and miss many other modes present in the data distribution. At its core, the generator is over-specialized for a given discriminator, which is stuck in a local minimum.

Moreover, training of a GAN is notoriously unstable [204]. While the non-saturating loss in Eq. 3.53 remedies the vanishing gradient problem early in the training, the discriminator feedback still gets less meaningful over time and hence the generator might collapse. In particular, the desired Nash equilibrium displays a saddle point that is difficult to find numerically [205, 201]. In addition, detecting convergence of a GAN is difficult as a universal metric for the fidelity of samples synthesized by the generator is missing [206].

Wasserstein Distance: Discriminator Estimates Transport Cost

Various variants of GANs have been developed to tackle the above mentioned challenges. A promising route is to improve the objective function of a GAN. A popular variant is the *Wasserstein GAN* (WGAN) that uses the *Earth Mover distance* (EMD) to measure the distance between $\mathcal{X}$ and $g_\Theta(\mathcal{Z})$ [204]. According to [204], the EMD has some appealing properties compared to other probability distance functions, such as relative entropy, cross entropy or Jensen-Shannon divergence: If the distributions have disjoint support, the aforementioned distance measurements yield gradients that are always zero. This is a major concern when dealing with real-world high-dimensional data sets, as the manifold hypotheses states that most of the probability mass is concentrated in lower-dimensional manifolds. Therefore, it is likely that the intersection of two probability distributions vanishes. The EMD circumvents these issues and guarantees continuity and differentiability. As such, the discriminator can be trained until optimality using the EMD without vanishing gradients and without getting stuck in local minima.

The EMD is defined as

$$W(p_\mathcal{X}, p_\Theta) = \inf_{\gamma\in\Gamma(p_\mathcal{X}, p_\Theta)} \mathbb{E}_{(\mathbf{x},\mathbf{x}')\sim\gamma}\big[\|\mathbf{x}-\mathbf{x}'\|\big], \tag{3.54}$$

where $\Gamma(p_{\mathcal{X}}, p_\Theta)$ denotes the set of all joint distributions $\gamma(\mathbf{x}, \mathbf{x}')$ whose marginals are $p_{\mathcal{X}}$ and p_Θ, respectively. $\gamma(\mathbf{x}, \mathbf{x}')$ can be interpreted as a transport plan indicating how much probability mass has to be moved from $\mathbf{x}$ to $\mathbf{x}'$ in order to make the two distributions match. As such, the EMD seeks the minimal transport cost.

The formulation of the EMD in Eq. 3.54 is highly intractable and most practical implementations apply an equivalent formulation of the EMD known as *Kantorovich-Rubinstein duality*:

$$W(p_{\mathcal{X}}, p_\Theta) = \max_{f \in \mathrm{Lip}(f) \leq 1} \mathbb{E}_{\mathbf{z} \sim \mathcal{Z}} \Big[f\big(g_\Theta(\mathbf{z})\big) \Big] - \mathbb{E}_{\mathbf{x} \sim \mathcal{X}} \Big[f(\mathbf{x}) \Big] \tag{3.55}$$

In Eq. 3.55, the maximum is taken over all functions $f : \mathbb{R}^D \to \mathbb{R}$ that are 1-Lipschitz. In practice, the function f is approximated with a NN c_Ψ and additional constraints are applied to ensure the 1-Lipschitz continuity. The mini-max game for the WGAN approach can then be written as

$$\min_\Theta \max_\Psi \mathbb{E}_{\mathbf{z} \sim \mathcal{Z}} \left[c_\Psi \big(g_\Theta(\mathbf{z}) \big) \right] - \mathbb{E}_{\mathbf{x} \sim \mathcal{X}} \left[c_\Psi(\mathbf{x}) \right]. \tag{3.56}$$

Various different methods exist to ensure the 1-Lipschitz continuity. A popular approach is *gradient penalty* (GP) that introduces an additional term to the cost function of the critic [207]. A differentiable function is one-Lipschitz if and only if it has gradients everywhere with norm at most one. A soft version of this constraint can be enforced by a penalty on the gradient norm

$$\mathcal{C}_{\mathrm{gp}}(c_\Psi) = \mathbb{E}_{\bar{\mathbf{x}} \sim \bar{\mathcal{X}}} \left[\left(||\nabla_{\bar{\mathbf{x}}} c_\Psi(\bar{\mathbf{x}})||_2 - 1 \right)^2 \right], \tag{3.57}$$

where $\bar{\mathbf{x}}$ is interpolated linearly between pairs of points $\mathbf{x}$ and $g_\Theta(\mathbf{z})$.

Chapter 4

Methodology of Deepbackmap: Adversarial Reverse-mapping of Condensed-phase Molecular Structures

In this chapter, *deepbackmap* (DBM) is introduced: A new method to tackle the backmapping problem for molecular structures. The method is based on a ML model that learns the coarse-to-fine mapping from training examples. Unlike other backmapping schemes, DBM aims at directly predicting equilibrated molecular structures that resemble the Boltzmann distribution. As such, the method does not rely on further energy minimization for relaxation and MD simulations for equilibration of the reverse-mapped structures. As illustrated in Fig. 4.1, a key feature of DBM is its applicability to condensed-phase molecular systems.

DBM is a deep generative model (DGM) trained with the generative adversarial approach. The training data consists of pairs of corresponding coarse-grained (CG) and fine-grained (FG) molecular structures. In particular, the CG structure is treated as a conditional variable for the generative process. Moreover, the ML model

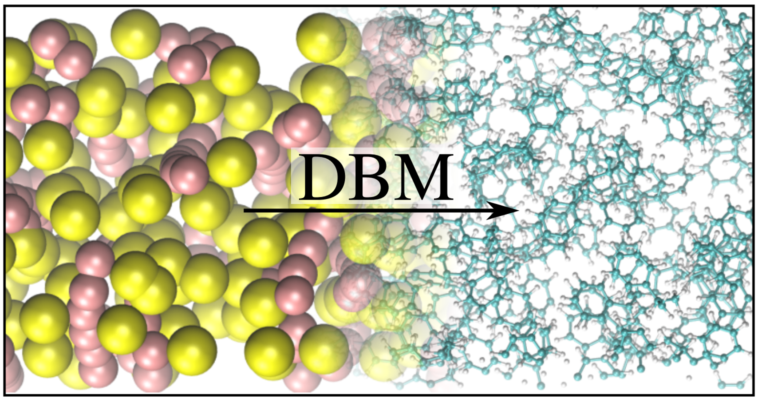

FIGURE 4.1: DBM generates Boltzmann-equilibrated atomistic structures conditional on the CG configuration using an adversarial network. It is designed for the backmapping of a condensed-phase molecular systems, such as polystyrene melts. Reprinted from [208].

is based on a convolutional neural network (CNN) architecture. As such, a regular discretization of 3D space is required, which prohibits scaling to larger spatial structures. Therefore, the generator is combined with an autoregressive approach that reconstructs the FG structure incrementally, i.e. atom by atom. The autoregressive reconstruction splits the backmapping task into a sequence of less complex tasks and thereby enables a local environment representation, i.e. in each step only local information is used. The locality of DBM is not only essential for the scalability of the model, but it is also a key feature to achieve remarkable transferability properties.

This chapter presents content that has been previously published in the following research articles. The content is reproduced here with kind permission from the other authors and the corresponding journals published this work.

Marc Stieffenhofer, Michael Wand, Tristan Bereau
Adversarial reverse mapping of equilibrated condensed-phase molecular structures
Machine Learning: Science and Technology, Volume 1, Number 4
DOI: 10.1088/2632-2153/abb6d4
© IOP Publishing Ltd, 2020

Marc Stieffenhofer, Tristan Bereau, Michael Wand
Adversarial reverse mapping of condensed-phase molecular structures: Chemical transferability
APL Materials 9, Volume 9, Number 3
DOI: 10.1063/5.0039102
© AIP Publishing LLC, 2021

4.1 Notation and Problem Formulation

Backmapping is the reintroduction of details along the CG degrees of freedom. More specifically, the backmapping function g is required to generate new coordinates $\mathbf{r} \in \mathbb{R}^{3n}$ for the n atoms in the system. As described in Sec. 2.4, g is a function of the coordinates $\mathbf{R} \in \mathbb{R}^{3N}$ of the N CG beads. Moreover, DBM incorporates additional information to improve the quality and transferability of the mapping. In particular, additional information is used to characterize the specific chemistry of both, the CG as well as the target FG structure.

Formally, let $\mathcal{A} = \{\mathbf{A}_I = (\mathbf{R}_I, \mathbf{C}_I) | I = 1, \ldots, N\}$ denote a snapshot of the CG system consisting of N beads. Each bead has position $\mathbf{R}_I \in \mathbb{R}^3$ and an associated type $\mathbf{C}_I \in \mathbb{R}^T$. The type $\mathbf{C}_I$ is expressed as a T dimensional one-hot vector, where T is the number of bead types, and reflects various chemistry specific attributes, such as the bead mass, the connectivity or associated force field parameters. Similarly, let $a = \{\mathbf{a}_i = (\mathbf{r}_i, \mathbf{c}_i) | i = 1, \ldots, n\}$ denote an atomistic snapshot of the system consisting of n atoms. Each atom $\mathbf{a}_i$ has position $\mathbf{r}_i \in \mathbb{R}^3$ and type $\mathbf{c}_i \in \mathbb{R}^t$, where t is the number of atom types and $\mathbf{c}_i$ is a one-hot vector. Each CG bead $\mathbf{A}_I$ has an associated set of atoms $\pi_I = \{\mathbf{a}_j | j \in \psi_I\} \subset a$, where ψ_I is the corresponding set of atom indices. Conversely, each atom $\mathbf{a}_i$ has an associated CG bead $\mathbf{A}_{\Psi_i}$, where Ψ_i denotes the index of the CG bead that atom $\mathbf{a}_i$ belongs to. The joint distribution of CG and FG snapshots is denoted with $\mathcal{X}$. In the following, a tuple $(\mathbf{x}_1, \ldots, \mathbf{x}_k)$ is represented as $\mathbf{x}_1^k$, where the subscript and superscript denote the indices for the first and the last element of the sequence, respectively.

DBM is a DGM designed to infer the conditional probability $p_{\mathcal{X}}(\mathbf{r}_1^n | \mathbf{A}_1^N, \mathbf{c}_1^n)$ from training data $\mathcal{D} = \{(\mathcal{A}_j, a_j)\}$ that consists of pairs of corresponding CG and FG snapshots. The conditional probability of the model $p_\Theta(\mathbf{r}_1^n | \mathbf{A}_1^N, \mathbf{c}_1^n)$, where Θ are the model parameters, is not inferred explicitly, but implicitly defined through a sampler g_Θ. More specifically, the sampler

$$g_\Theta : \mathbb{R}^{3N}, \mathbb{R}^{TN}, \mathbb{R}^{tn} \rightarrow \mathbb{R}^{3n} \tag{4.1}$$

generates a list of coordinates $g_\Theta(\mathbf{A}_1^N, \mathbf{c}_1^n) = \mathbf{r}_1^n$. The overall goal is to tune the parameters Θ of g_Θ such that $p_\Theta \approx p_{\mathcal{X}}$.

4.2 Autoregressive Reconstruction

Direct sampling from $p_{\mathcal{X}}(\mathbf{r}_1^n | \mathbf{A}_1^N, \mathbf{c}_1^n)$ poses significant challenges. At first, the complexity of the sampling problem rises with the number of particles n. However, the size of molecular systems studied with computer simulations is typically large. As a consequence, a sampler g_Θ designated to generate all coordinates at once has to solve a problem with unreasonably large dimensionality. Such an one-shot approach is ultimately limited to rather small system sizes.

Moreover, direct sampling restricts the transferability of the trained model. As an example, the number of CG beads and atoms is fixed in a one-shot model, i.e. the model is only applicable to systems of the same size. This implies that the data required for training needs to be as high-dimensional as the target system. Such a strategy is questionable, as the purpose of most multiscale approaches is to extend the accessible system size. A more progressive approach is to train the model on FG samples with rather small system sizes, but deploy it on larger CG structures. In addition, chemical transferability is limited in a one-shot approach, as the trained sampler expects to generate the same kind of molecules it was trained on. As such, transferring the learned correlation across chemical space is not straightforward.

The proposed method DBM circumvents these limitations by factorizing $p_{\mathcal{X}}$ in terms of atomic contributions. More precisely, the generation of one specific atom becomes conditional on both, the CG beads as well as all the atoms previously reconstructed. Such a factorization can be obtained by applying the chain rule for

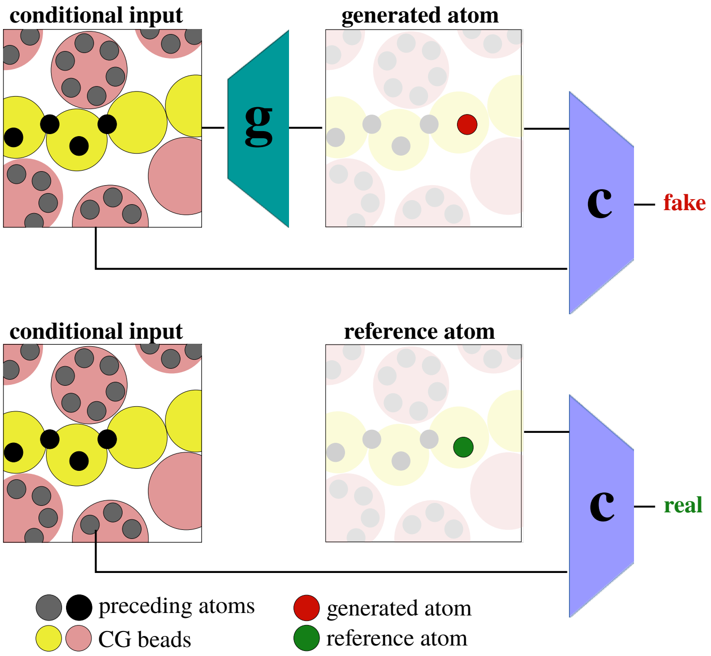

FIGURE 4.2: Adversarial autoregressive approach: The generator, g_Θ, sequentially samples atom positions conditional on the CG structure and the existing atoms. A critic network, c_Ψ, estimates the discrepancy between reference and generated atoms. Reprinted from [208].

probabilities

$$p_{\mathcal{X}}(\mathbf{r}_1^n|\mathbf{A}_1^N, \mathbf{c}_1^n) = \prod_{i=1}^{n} p_{\mathcal{X}}(\mathbf{r}_{s(i)}|\mathbf{r}_{s(1)}^{s(i-1)}, \mathbf{c}_{s(1)}^{s(i)}, \mathbf{A}_1^N), \tag{4.2}$$

where s sorts the atoms in the order of reconstruction and $\mathbf{r}_{s(1)}^{s(i-1)}$ denotes the atoms that have already been generated. Specifically, $s(i)$ denotes the atom index at the ith position in the ordering. Eq. 4.2 splits a complex, high-dimensional problem into a sequence of rather simple tasks, namely to learn the conditionals $p_{\mathcal{X}}(\mathbf{r}_{s(i)}|\mathbf{r}_{s(1)}^{s(i-1)}, \mathbf{c}_{s(1)}^{s(i)}, \mathbf{A}_1^N)$. Such a modular reconstruction increases the flexibility of the model and offers a perspective to release the aforementioned limitations.

In this study, the conditionals are implicitly learned by a generative model g_Θ, i.e. g_Θ is trained to generate and refine atom coordinates sequentially. The local placement of the atoms is thereby learned with an adversarial approach, as illustrated in Fig. 4.2. The dependence on earlier predictions of g_Θ makes the method *autoregressive*.

4.2.1 Ordering of Molecular Graphs

The factorization proposed in Eq. 4.2 requires a strict ordering s of the particles. However, the ordering s is generally not unique and has to be defined artificially. Here, the order is defined on multiple levels: The ordering of molecules, as well as the ordering of beads and atoms within a molecule. While the ordering of the molecules is less important for the performance of the model, the traversal through the molecular structure has to be chosen carefully.

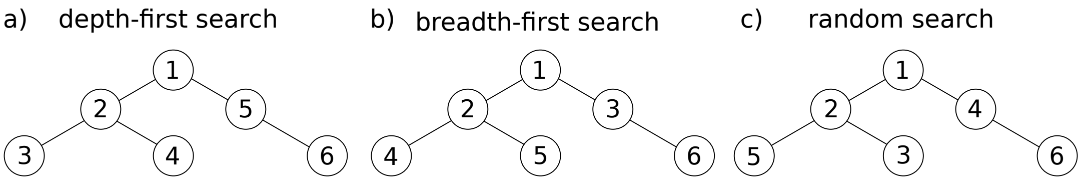

FIGURE 4.3: Three different options to traverse a graph: a) depth-first search, b) breadth-first search, c) random search.

The algorithm DBM iterates through the sequence of molecules, which is arbitrarily chosen. Each molecule is completely reconstructed before the next molecule is visited. In order to sort the particles within each molecule, the molecular structure is represented as a graph. Specifically, particles and bonds are mapped to the nodes and edges of the graph. As such, the sorting of the particles is described as a graph traversal. Note, that molecular graphs are generally undirected and can be cyclic or acyclic [209]. Therefore, the molecular graph has no specific sorting. Here, three different strategies are available to traverse the graph. In each strategy, a root node is selected from which the traversal origins. If the structure is linear, the ends are typically chosen. From there on, the subsequent nodes are selected according to one of the following search-algorithms:

- *depth-first-search*: Each branch of the graph is explored as far as possible before backtracking. See Fig. 4.3 a).
- *breadth-first-search*: All nodes at the present depth are explored before moving on to the nodes at the next depth level. See Fig. 4.3 b).
- *random*: The subsequent node is chosen randomly. See Fig. 4.3 c).

Practice has shown that an ordering based on the depth-first-search yields the best performance regarding the quality of reconstructed molecules.

DBM sorts the atoms depending on both, the CG as well as the atomistic molecular topology: In an outer loop, the CG molecular graph is explored yielding a sorting for the beads. Within each bead $\mathbf{A}_I$, DBM iterates through the fragment π_I before visiting subsequent beads.

Note that it is possible to let DBM learn the order of reconstruction itself. However, while such an approach is feasible for small molecules, it poses significant computational and conceptual challenges for large molecular structures. In particular, learning the ordering of a molecule requires a cost-function that enables backpropagation of the error signal for every step in order to find the best reconstruction strategy. This becomes intractable for large molecules, since unrolling of the recursive approach requires to store a copy of the model in memory for each step (see Sec. 3.3.2). Moreover, obtaining a suitable representation is more complicated (see Sec. 4.3), as it requires to represent the entire molecular structure in every step, which again limits the scalability of the model. As a remedy, the algorithm would have to automatically adapt to a local environment centered around the current focus of interest.

4.2.2 Initial Structure with Forward Sampling

The first step of the proposed algorithm is to generate an initial structure. To this end, *forward sampling* is applied based on the factorization in Eq.4.2 [210]. The algorithm starts by sampling the variables with no parents from a prior distribution, i.e. the atom position $\mathbf{r}_{s(1)}$ for the first atom in the ordering s. Note that even this first atom position is not arbitrary, since translational symmetry is lost by conditioning on the CG structure. Subsequent variables $\mathbf{r}_{s(i)}$ are generated by sampling from the conditional probability distributions $p_\Theta(\mathbf{r}_{s(i)}|\mathbf{r}_{s(1)}^{s(i-1)}, \mathbf{c}_{s(1)}^{s(i)}, \mathbf{A}_1^N)$ given the atoms generated in the previous steps.

Forward sampling yields accurate results if the underlying graph structure has a *topological order*, i.e. a graph traversal in which each node is visited only after all of its dependencies are explored [210]. Note, that a topological order exists only for directed acyclic graphs, which is generally not the case for molecular graphs. As a consequence, forward sampling applied to molecular graphs can yield structures with low statistical weight, as it requires to sample some variables for which crucial information might be missing. In other words, it is not possible to find the optimal position for an atom without knowing its environment. This issue becomes especially apparent when the underlying graph contains cyclic structures, such as the phenyl rings in polystyrene. In general, the autoregressive approach is prone to accumulate errors even without cyclic structures, since misplaced atoms can always affect the placement of subsequent atoms.

4.2.3 Refinement with Gibbs Sampling

As outlined above, accurate sampling of molecular structures calls for more feedback than a simple forward sampling strategy allows. This is especially true for condensed-phase systems, where great care has to be taken to avoid steric clashes. To this end, a variant of *Gibbs sampling* is applied, which subsequently refines the initial molecular structures [211].

Gibbs sampling is a Markov chain Monte Carlo algorithm. As such, it constructs a Markov Chain that eventually converges towards the target distribution. Gibbs sampling starts from an initial structure $\mathbf{r}_1^{n^{[0]}}$ and resamples each component iteratively along the ordering s. Importantly, each further iteration still updates a single component at a time, but each component is conditioned on *all* other components: The component $\mathbf{r}_{s(i)}^{[k+1]}$ is conditioned on the values $\mathbf{r}_{s(1)}^{s(i-1)^{[k+1]}}$ of already updated components at the current step $k+1$ up to $s(i)$ and thereafter, the values $\mathbf{r}_{s(i+1)}^{s(n)^{[k]}}$ from the previous step k are used. More precisely, $\mathbf{r}_{s(i)}^{[k+1]}$ is sampled according to $p_\Theta(\mathbf{r}_{s(i)}^{[k+1]}|\mathbf{r}_{s(1)}^{s(i-1)^{[k+1])}}, \mathbf{r}_{s(i+1)}^{s(n)^{[k]}}, \mathbf{c}_{s(1)}^{s(n)}, \mathbf{A}_1^N)$. Experiments confirmed that such Gibbs sampling leads to a good approximation of the target distribution $p_{\mathcal{X}}$, even with a small number of iterations.

4.3 Representation of Molecular Structures

Learning of complex, high-dimensional and higher-order dependencies in generative models is a hallmark of computer vision. As outlined in Sec. 3.3.3, one of the most successful learning algorithms for processing image content are deep CNNs [212, 213, 214, 215, 6]. Their success relies on their ability to exploit spatial and temporal correlations. Other key attributes of CNNs are automatic feature extraction, hierarchical learning and parameter sharing [216].

In order to leverage modern CNNs for the backmapping task, an explicit spatial discretization of ambient space is required. Similar to pixels in a two dimensional image, the three dimensional molecular structure has to be mapped onto a voxel-based representation [217]. To this end, atoms and beads are represented as smooth densities, γ and Γ, respectively. More specifically, Gaussian distributions are used to model particle densities: An atom $\mathbf{a}_i$ at position $\mathbf{r}_i$ is represented as

$$\gamma_i = \exp\Big(-\frac{(\mathbf{x}-\mathbf{r}_i)^2}{2\sigma^2}\Big), \tag{4.3}$$

where $\mathbf{x}$ is a spatial location in Cartesian coordinates, expressed on a discretized grid due to voxelization, and σ is the Gaussian width, which is treated as a hyper parameter. The same concept is used to represent CG beads.

4.3.1 Local Environment

The proposed voxel-based representation is well suited for deploying CNNs. However, it does not adapt well to large molecular structures, as the computational cost scales with the cubic grid size. To circumvent these limitations, the autoregressive approach is used to build-up larger structures incrementally, while restricting the receptive field of the CNN: Rather than representing the molecular structure as a whole, the model becomes conditional on *local environments*, where the information is limited to a cutoff r_{cut}. Such a locality assumption makes the model scalable to larger system sizes, i.e. the computational cost scales linearly with the number of FG particles.

Beside introducing a cutoff r_{cut}, the local environments are centered and aligned. This improves generalization, as translational and rotational degrees of freedom are removed. In other words, the ML algorithm does not have to learn the corresponding equivariance from (additional) training data. Note that the regular CNNs deployed in this thesis are equivariant with respect to translations by construction, but not with respect to rotations. Although promising progress has been achieved recently regarding the design of rotational equivariant networks, it is not straightforward to extend these approaches to generative models [218, 219, 220]. It is therefore beneficial to make use of the given molecular geometry to reduce the rotational degrees of freedom. Experiments confirm that the alignment of the local environment improves the performance of the model significantly.

Specifically, the local environment ϵ_i for an atom $\mathbf{a}_i$ is centered around the current bead of interest $\mathbf{A}_{\Psi_i}$, i.e. all atoms and beads are shifted around $\mathbf{R}_{\Psi_i}$. The local environment contains the densities of all particles within a cubic environment of size $2r_{\text{cut}}$. Further, the local environment is rotated to a local axis. To this end, the bond between the current CG bead Ψ_i and its predecessor is aligned to the local z axis by a rotation matrix $\mathbf{M}_{\Psi_i}$. This yields the definition for the local environment

$$\epsilon_i(\mathbf{x}) = \sum_{j=1, j \neq i}^{n} \gamma_j(\mathbf{M}_{\Psi_i}(\mathbf{x} - \mathbf{R}_{\Psi_i})) \tag{4.4}$$

$$+ \sum_{J=1}^{N} \Gamma_J(\mathbf{M}_{\Psi_i}(\mathbf{x} - \mathbf{R}_{\Psi_i})), \tag{4.5}$$

which extends over the region $-r_{\text{cut}} < x_\alpha < r_{\text{cut}}$, where α runs over the three Cartesian coordinates. Note that $\mathbf{x}$ is discretized over a regular grid. In practice, r_{cut} is chosen such that several beads are present in each local environment.

In the case of forward sampling, an incomplete representation $\tilde{\epsilon}_i(\mathbf{x})$ has to be used. In particular, $\tilde{\epsilon}_i(\mathbf{x})$ excludes all atoms $\mathbf{a}_{s(j)}$ for which $j \geq s^{-1}(i)$, where $s^{-1}(i)$ denotes the position of the atom $\mathbf{a}_i$ in the ordering s,

$$\tilde{\epsilon}_i(\mathbf{x}) = \sum_{j=1}^{s^{-1}(i)-1} \gamma_{s(j)}(\mathbf{M}_{\Psi_i}(\mathbf{x} - \mathbf{R}_{\Psi_i})) \tag{4.6}$$

$$+ \sum_{J=1}^{N} \Gamma_J(\mathbf{M}_{\Psi_i}(\mathbf{x} - \mathbf{R}_{\Psi_i})). \tag{4.7}$$

Centering and alignment of ϵ_i and $\tilde{\epsilon}_i$ removes three translational and two rotational degrees of freedom. This leaves one rotational degree of freedom around the director axis, which the model is supposed to learn from the training data. For this reason, the training set is augmented by means of rotations around the director axis.

4.3.2 Feature Embedding

The input of CNNs is typically a two or three dimensional image composed of multiple *feature channels*, i.e. each pixel or voxel is vector-valued. The different channels provide different views on the data. As an example, an RGB image contains three separate channels: One feature channel for every primary color. Similarly, the input for DBM is composed of multiple channels to encode the presence of atoms and beads of a certain type. In the most basic version, the representation given in Eq. 4.4 is used directly, which yields a single feature channel encoding all atoms and beads.

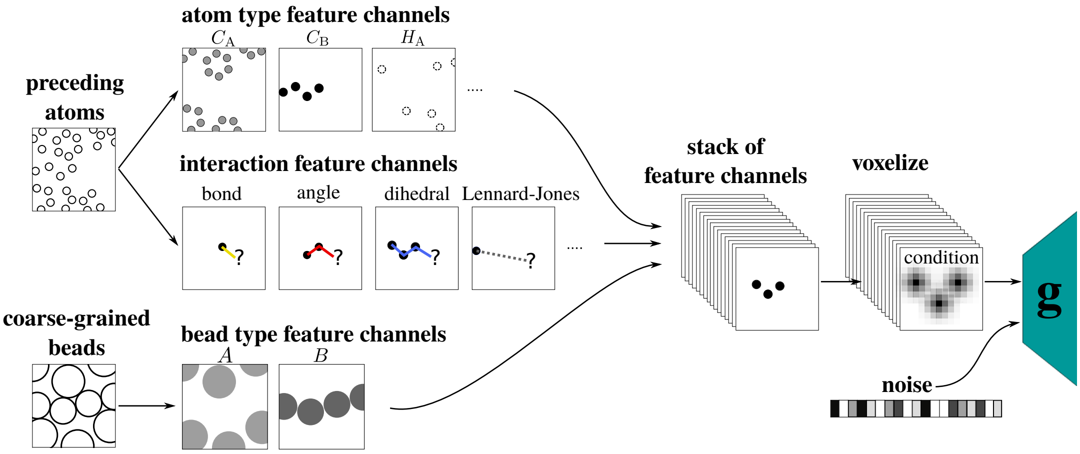

FIGURE 4.4: Feature embedding of a local environment. Atoms and CG beads of the local environment are split into separate channels according to their atom/bead types. In addition, the atomic environment of a current atom of interest is split in terms of molecular interactions that distinguish between bond, bending angle, torsion or Lennard-Jones. Afterwards, all channels are voxelized. The final input for the generator network g_Θ consists of the voxelized feature channels and an additional noise sample. Reprinted from [208].

However, this leads to overlapping atom and bead densities that deteriorate the spatial resolution of the model. Moreover, a single feature channel does not take the different types of atoms and beads into account and therefore, important information is lost. The opposite extreme is to assign each atom or bead to a separate feature channel. Such a representation is not flawless as well, because the permutational invariance of the atoms and beads is lost, i.e. atoms and beads have to be presented in a fixed order, which reduces the generalization ability of the model dramatically.

As shown in Fig. 4.4, various feature channels are created to improve the representation defined in Eq. 4.4. Each channel reflects a different attribute of the atoms and beads assigned to it. For example, an attribute can encode the chemical element or represent the set of force-field parameters associated with a specific atom type. Further, attributes can encode the functional form of the interaction to the current atom of interest. Interaction types distinguish between bond, bending angle, torsion or Lennard-Jones. At its core, such interaction attributes reflect the local structure of the molecular graph, as they represent short paths with one (bond), two (bending angle), three (torsion) or more (Lennard-Jones) edges originating from the current atom of interest. As such, DBM is trained to place an atom that completes the given paths. Moreover, paths of the same length can be split up further into distinct feature channels in order to emphasize the difference of their associated force-field parameters. For example, a bending angle $C - C - C$ between carbon atoms might be treated differently than a bending angle $H - C - H$ between carbon and hydrogen atoms.

Formally, let $f \in \{1, 2, \ldots, N_f\}$ denote the index of the N_f different feature channels. The activation function, $h_f(\mathbf{a}_j; \mathbf{a}_i)$, is defined to denote association of an atom

$\mathbf{a}_j$, which is present in the local environment of the current atom of interest $\mathbf{a}_i$, with a channel f

$$h_f(\mathbf{a}_j;\mathbf{a}_i) = \begin{cases} 1, & \text{if atom } \mathbf{a}_j \text{ has feature } f \text{ (with respect to } \mathbf{a}_i) \\ 0, & \text{otherwise.} \end{cases} \tag{4.8}$$

Note, that some attributes, such as the associated atom types, have no dependence on the current atom of interest $\mathbf{a}_i$. Other attributes, like the interaction channels, have to be defined with respect to $\mathbf{a}_i$. In addition, an activation function $H_f(\mathbf{A}_J)$ is defined to encode attributes of the CG beads, i.e. the bead types. In summary, the following featurized representation is obtained

$$\epsilon_i(\mathbf{x}, f) = \sum_{j=1, j\neq i}^{n} \gamma_j(\mathbf{M}_{\Psi_i}(\mathbf{x} - \mathbf{R}_{\Psi_i}))h_f(\mathbf{a}_j;\mathbf{a}_i) \tag{4.9}$$

$$+ \sum_{J=1}^{N} \Gamma_J(\mathbf{M}_{\Psi_i}(\mathbf{x} - \mathbf{R}_{\Psi_i}))H_f(\mathbf{A}_J). \tag{4.10}$$

The featurized representation for the forward-sampling $\tilde{\epsilon}_i$ is constructed similarly.

4.4 Conditional Generative Adversarial Network

The autoregressive approach turns the complex problem of generating molecular structures into a sequence of much simpler decisions. However, learning the local placement of the atoms is still a challenging task. Implementing a rule based decision algorithm, for example grounded on the geometry or energy of the structure, quickly becomes tedious and problem specific. Even more importantly, such methods would not be able to reproduce the desired Boltzmann distribution. On the other hand, ML models have shown the ability to learn complex distributions without relying on tedious rule based programming, since decisions are learned from training data.

At this point, the engine of DBM is introduced: A ML model to learn the local placement of the atoms. The recent success of generative adversarial networks (GANs) in generating sharp, photorealistic images has motivated the application of the adversarial training approach for this task. As stated in Sec. 3.4.3, a generator g_Θ with parameters Θ maps samples $\mathbf{z} \in \mathbb{R}^d$ from a latent distribution $\mathcal{Z}$ into the ambient space $\mathbb{R}^D$. A second model, the discriminator c_Ψ with parameters Ψ, acts as a distance measure in ambient space $\mathbb{R}^D$ for the real distribution $\mathcal{X}$ and the distribution of synthetic samples $g_\Theta(\mathcal{Z})$. However, the standard GAN approach does not offer much control over the generative process, as the correlations between the latent and generated distribution are essentially arbitrary. On the other hand, reverse-mapping requires to condition the generative process on the CG structure. Moreover, the autoregressive approach requires to provide the information of previously reconstructed atoms. To this end, a conditional GAN (cGAN) approach is used: Both networks g_Θ and c_Ψ are provided with auxiliary information to generate samples related to this additional input. As such, the discriminator does not only evaluate the quality of the generated structure alone, but also its consistency with the given auxiliary information. In the present work, the conditional input for both networks consists of the local environment representation ϵ_i and the atom type $\mathbf{c}_i$, denoted with $\mathbf{u}_i = (\epsilon_i, \mathbf{c}_i)$.

4.4.1 Densities and Coordinates

The CNN architecture of the critic network c_Ψ requires that the prediction of the generator g_Θ has a smooth density representation to perform adversarial training. At the same time, the position of the atom has to be expressed ultimately as a point coordinate. Two options are available to generate both consistently:

1) The generator g_Θ predicts a smooth-density representation $\hat{\gamma}_i := g_\Theta(\mathbf{z}, \mathbf{u}_i)$, which is collapsed back to a point coordinate $\hat{\mathbf{r}}_i$. To this end, a weighted average is computed, discretized over the voxel-grid

$$\hat{\mathbf{r}}_i = \frac{1}{w} \sum_{m,k,l=1}^{d} x_{mkl} \hat{\gamma}_i(x_{mkl}), \tag{4.11}$$

where $w = \sum_{m,k,l=1}^{d} \hat{\gamma}_i(x_{mkl})$ is a normalization constant and x_{mkl} a particular coordinate value within the three-dimensional grid of size d.

2) The generator g_Θ directly predicts a point coordinate $\hat{\mathbf{r}}_i := g_\Theta(\mathbf{z}, \mathbf{u}_i)$, which is mapped to a smooth-density representation $\hat{\gamma}_i$. To this end, a Gaussian mapping is used

$$\hat{\gamma}_i = \exp\Big(- \frac{(\mathbf{x} - \hat{\mathbf{r}}_i)^2}{2\sigma^2} \Big). \tag{4.12}$$

Experiments have shown that both versions perform equally well. If not stated otherwise, the first option is used in the following. In either case, both $\hat{\gamma}_i$ as well as $\hat{\mathbf{r}}_i$ are differentiable and thus can be easily incorporated in a cost function.

4.4.2 Adversarial Cost Function for Training on Sequences

Training of the networks is based on the Wasserstein GAN protocol described in Sec. 3.4.3, which is extended to incorporate conditional information. However, optimal positioning of an atom is only possible if the previous atoms are placed correctly. In other words, the autoregressive reconstruction of the molecular structure is prone to accumulate errors. Therefore, the training protocol of the network has to take the autoregressive nature of the approach into account. More specifically, the generator has to be penalized for its actions in the past if they hinder the correct placement of the current atom. To this end, backpropagation of the error signal is applied to an entire sequence of generated atom positions.

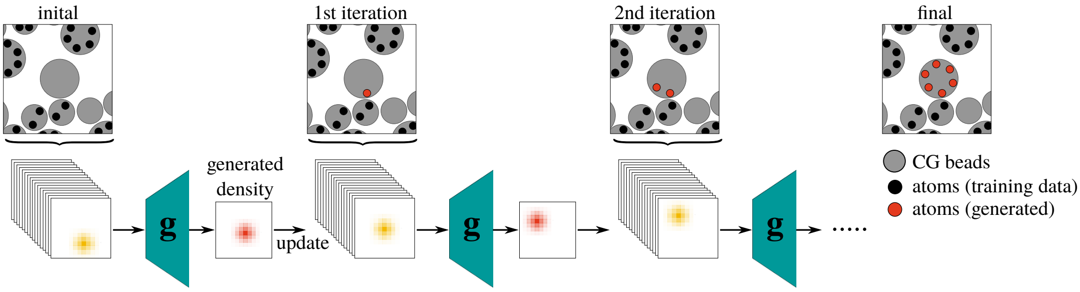

FIGURE 4.5: Autoregressive training. Starting from an atomistic configuration taken from training data (black) the predicted atoms (red) will be added to the local environment description for predicting the next atom in the sequence. Reprinted from [208].

For simplicity and practical reasons, the training sequences ω_I contain the indices ψ_I of atoms corresponding to a single CG bead $\mathbf{A}_I$. Unlike the set ψ_I, the sequence ω_I is ordered according to s. The autoregressive adversarial cost-function $\mathcal{C}_{\text{ar}}$ the generator g_Θ aims to minimize is expressed as

$$\min_{\Theta} \mathcal{C}_{\text{ar}}(g_\Theta) = \min_{\Theta} \mathbb{E}_I \Big[\frac{1}{|\omega_I|} \sum_{i \in \omega_I} c_\Psi(\mathbf{u}_i, g_\Theta(\mathbf{z}, \mathbf{u}_i)) \Big], \tag{4.13}$$

where $|\omega_I|$ is the number of atoms in the sequence. The critic c_Ψ is trained to minimize

$$\min_{\Psi} \mathcal{C}_{\text{ar}}(c_\Psi) = \min_{\Theta} \mathbb{E}_I \Big[\frac{1}{|\omega_I|} \sum_{i \in \omega_I} c_\Psi(\mathbf{u}_i, \gamma_i) - c_\Psi(\mathbf{u}_i, g_\Theta(\mathbf{z}, \mathbf{u}_i)) + \lambda_{\text{gp}} \mathcal{C}_{\text{gp}}(\mathbf{u}_i, \bar{\gamma}_i) \Big], \tag{4.14}$$

where $\mathcal{C}_{\text{gp}}$ is the gradient penalty to enforce the 1-Lipschitz continuity of the critic, as explained in Sec. 3.4.3. The prefactor λ_{gp} scales the weight of the gradient penalty and is set to $\lambda_{\text{gp}} = 10$ in all experiments. The density $\bar{\gamma}_i$ is interpolated linearly between pairs of points γ_i and $g_\Theta(\mathbf{z}, \mathbf{u}_i)$.

As illustrated in Fig. 4.5, the local environments presented to the network during training are composed of atoms taken from the training data as well as already generated atoms: The initial local environment for the first atom in a sequence ω_I is constructed from training data. After each step, the generated density $\hat{\gamma}_i$ is added to the local environment representation for the next atom in the sequence, until all atoms in the sequence are generated. As such, the computational graph for the sequence generation consists of $|\omega_I|$ copies of the generator. The critic judges each step and the error signals of all $|\omega_I|$ steps are accumulated. After a sequence is completed, the accumulated cost is backpropagated through the unrolled network. As such, this approach takes dependencies among the different steps into account.

4.5 Potential Energy as Regularizer

Ideally, the adversarial cost is already sufficient to drive the generator towards reproducing the desired Boltzmann distribution. However, training of a GAN is notoriously unstable and the parameters of a GAN easily diverge. As a result, GANs have a number of failure modes, such as mode-collapse or failure to converge (see Sec. 3.4.3). Various forms of regularization have been deployed to address those issues and improve generalization, including gradient penalties, weight normalization or architectural methods [221, 222].

Unlike data sets commonly used in the ML community, the target distribution $p_{\mathcal{X}}(\mathbf{x}) \propto \exp\left[\frac{U(\mathbf{x})}{k_b T}\right]$ for the desired molecular structures is already known up to a normalization constant, i.e. the partition function. This knowledge can be incorporated in the training of DBM to improve its performance and to monitor the training process. Specifically, the potential energy U of generated structures is utilized as an additional term $\mathcal{C}_{\text{pot}}$ in the cost function of the generator. As such, $\mathcal{C}_{\text{pot}}$ acts as a regularizer that effectively narrows down the functional space of the generator by penalizing structures with high potential energy. In Bayesian terms, $\mathcal{C}_{\text{pot}}$ incorporates prior believe about the model into the training, as it helps steering the optimization towards generating structures with high Boltzmann weight. It thereby effectively accelerates convergence and helps to improve accuracy.

$\mathcal{C}_{\text{pot}}$ depends on the set of atoms π_I corresponding to the current CG bead of interest $\mathbf{A}_I$, as well as reference atoms $N_I = \{\mathbf{a}_j | \mathbf{a}_j \in \pi_J, J \neq I, |\mathbf{R}_J - \mathbf{R}_I| < r_{\text{cut}}\}$ in the local environment of $\mathbf{A}_I$ that are associated to different beads. In the following, e_t denotes the potential energy of specific intra- and intermolecular interactions, which are described in Sec. 2.2.2. Specifically, t runs over the interaction types: bond, angle, dihedral, and non-bonded Lennard-Jones.

In this study, two different cost functions based on the potential energy are used. The first prior cost function, $\mathcal{C}^{(1)}_{\text{pot}}$, aims at minimizing the potential energy of generated structures,

$$\mathcal{C}^{(1)}_{\text{pot}}(\hat{\pi}_I, N_I) = \sum_t \lambda_t e_t(\hat{\pi}_I, N_I), \tag{4.15}$$

where $\hat{\pi}_I$ denotes atoms positioned by the generator g_Θ and λ_t scales the different energy terms. The second prior cost function, $\mathcal{C}^{(2)}_{\text{pot}}$, penalizes discrepancies between potential energies of generated and reference structures,

$$\mathcal{C}^{(2)}_{\text{pot}}(\hat{\pi}_I, \pi_I, N_I) = \sum_t \lambda_t |e_t(\hat{\pi}_I, N_I) - e_t(\pi_I, N_I)|. \tag{4.16}$$

Overall, the following cost function is minimized by the generator

$$\min_\Theta \mathcal{C}_{\text{tot}}(g_\Theta) = \min_\Theta \mathbb{E}_I \Big[\frac{1}{|\omega_I|} \sum_{i \in \omega_I} c_\Psi(\mathbf{u}_i, g_\Theta(\mathbf{z}, \mathbf{u}_i)) + \lambda_{\text{pot}} \mathcal{C}_{\text{pot}}(\hat{\pi}_I, \pi_I, N_I)\Big]. \tag{4.17}$$

Note that $\mathcal{C}_{\text{pot}}$ might be in a conflict with the adversarial cost function: While the purely data driven adversarial cost ultimately aims at reproducing the target distribution, the prior cost function encourages the generator to produce structures with a certain potential energy. In an extreme case, where optimization is solely based on the prior cost function, the generator is likely to collapse. Therefore, $\mathcal{C}_{\text{pot}}$ aims at supporting the adversarial optimization, which might suffer from resolution limits of the voxel representation, for fine tuning of the generated structures. As such, the prior cost function is scaled with an appropriately low weight λ_{pot}, such that it provides a significant contribution to the total cost only for high-energy structures. In particular, training starts with $\lambda_{\text{prior}} = 0.0$ and is increased slowly during the course of optimization.

4.6 Discussion

DBM is a new method based on ML for the reverse-mapping of molecular systems in the condensed-phase. The method is developed to avoid further energy minimization for relaxation and MD simulations for equilibration of the generated FG structures. Moreover, DBM requires little human intervention, since the reinsertion of local details is learned from training examples.

The generative adversarial approach is used to train DBM. To this end, a training set consisting of pairs of corresponding CG and FG molecular structures is used. While the target of the generator is to reproduce FG configurations, the CG structures are treated as conditional variables for the generative process. The generator g reinserts missing degrees of freedom along CG variables and a discriminator c compares the generated structures with the training examples. Since the input for the discriminator consists of both, the CG and the FG configuration, the discriminator evaluates not only the quality of the generated FG structure, but also its consistency with the given CG structure.

A CNN architecture is used for both models that requires a regular discretization of 3D space, which limits scaling to larger spatial structures. Therefore, the generator is combined with an autoregressive approach that reconstructs the FG structure incrementally, i.e. atom by atom. While DBM only learns local correlations, large-scale features are adapted from the CG structure. As such, only local information is required in each step, which makes the method scalable to larger system sizes. In addition, the local environment approach is a key feature for the generalizability of DBM, which will be explored in the subsequent chapter.

The order of reconstruction is defined by a traversal of the molecular graph. Since molecular graphs are generally undirected and can be cyclic or acyclic, the depth-first-search algorithm is applied to obtain a ordering for the atoms. In a first step, DBM generates atom positions with no parents and positions of subsequent atoms are based on the atoms generated in previous steps. However, such forward sampling only yields accurate results if the underlying graph structure has a topological order, i.e. a graph traversal in which each node is visited only after all of its dependencies are explored. As such, accurate sampling of molecular structures requires more feedback than a simple forward sampling strategy provides. To this end, a variant of Gibbs sampling is applied, which subsequently refines the initial molecular structures by iteratively resampling the atom positions. Each further iteration still updates one atom at a time, but uses the knowledge of all other atoms.

The potential energy function of the system can be incorporated in the training of DBM to improve its performance and to monitor the training process. Specifically, the potential energy U of generated structures is utilized as an additional term $\mathcal{C}_{\text{pot}}$ in the cost function of the generator. As such, $\mathcal{C}_{\text{pot}}$ acts as a regularizer that steers the optimization towards generating structures with high Boltzmann weight. It thereby effectively accelerates convergence and helps to improve accuracy.

Chapter 5

Performance and Transferability of DBM: Reverse-mapping of Syndiotactic Polystyrene

Deepbackmap (DBM) is a ML-based approach for the reverse-mapping of condensed-phase molecular structures. The approach sequentially reconstructs atomic environments. Moreover, the method is based on a locality assumption, i.e. the placement of one atom is assumed to rely only on short-range force field related features. Specifically, DBM learns to reproduce correlations in a local environment, while large-scale features are adapted from the coarse-grained (CG) structure. It can be hypothesized that such local environments strongly overlap across different state points and across chemical space. Therefore, the small-scale features learned by DBM are likely to generalize, which is examined in this chapter. In particular, DBM is applied to a challenging condensed-phase polymeric system that consists of syndiotactic polystyrene (sPS). The performance of DBM is analyzed in terms of three important aspects: 1) The general reverse-mapping capability of the model is probed, i.e. the ability to reproduce a reference all-atom (AA) distribution from CG configurations. 2) The transferability of the model across different thermodynamic state points is tested. To this end, DBM is trained solely on data obtained in a high-temperature melt. Afterwards, the model is transferred towards lower temperatures, where the system is in a crystalline state. 3) The transferability of DBM across chemical space is examined. In particular, DBM is trained on liquids of small molecules. After training, the model is applied to the more challenging polymeric system of sPS. In the following, each of the three aspects is addressed in a separate section. In the end of this chapter, the results of all sections are summarized and discussed.

This chapter presents content that has been previously published in the following research articles [208, 223]. The content is reproduced here with kind permission from the other authors and the journals which published these articles.

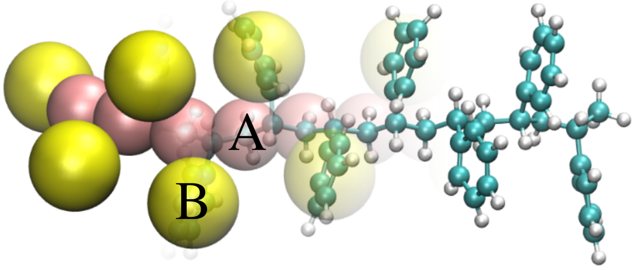

FIGURE 5.1: CG (left) and AA (right) representation of sPS. The CG monomer consists of two beads, denoted *A* for the chain backbone and *B* for the phenyl ring. Reprinted from [208].

Marc Stieffenhofer, Michael Wand, Tristan Bereau
Adversarial reverse mapping of equilibrated condensed-phase molecular structures
Machine Learning: Science and Technology, Volume 1, Number 4
DOI: 10.1088/2632-2153/abb6d4
© IOP Publishing Ltd, 2020

Marc Stieffenhofer, Tristan Bereau, Michael Wand
Adversarial reverse mapping of condensed-phase molecular structures: Chemical transferability
APL Materials 9, Volume 9, Number 3
DOI: 10.1063/5.0039102
© AIP Publishing LLC, 2021

5.1 Set-up and Reference Data

In this section, the reference data used to train and evaluate the performance of DBM is introduced. Moreover, specifications for DBM are given pertaining the training and inference procedure. In addition, a second backmapping protocol based on energy minimization (EM) is introduced as a baseline method.

5.1.1 Syndiotactic Polystyrene

Polystyrene (PS) is an aromatic polymer made from the monomer styrene, which is an organic compound consisting entirely of carbon and hydrogen. Physical and chemical properties of PS depend significantly on its tacticity, i.e. the arrangement of the phenyl groups along the polymer backbone. In this study, syndiotactic polystyrene (sPS) is used, where the phenyl groups are arranged on alternating sides of the polymer backbone. An illustration of a single polymer chain with AA as well as CG resolution is shown in Fig. 5.1.

Despite its simple chemical structure, sPS displays a rich conformational space and exhibits complex polymorphic behavior. As such, sPS is a well suited candidate to study the transferability properties of DBM. Upon thermal annealing, the sPS melt undergoes a phase transition from amorphous to a crystalline phase at $T \approx 450$ K [224]. Five different crystalline forms of sPS have been reported experimentally. Here, the focus is set on the α and β polymorphs, which are illustrated in Fig. 5.4.

The atomistic data in this study is reported in Liu *et al.* [224]; the underlying force field is based on the work of Mueller-Plathe [225]. The system is sampled using Replica Exchange MD simulations, which are performed using the molecular dynamics package GROMACS 4.6 [226]. The simulations are carried out in the NPT ensemble using the velocity rescaling thermostat and the Parrinello-Rahman barostat. An integration time step of 1 fs is used. For additional details regarding the simulations of the sPS system, the reader is referred to the work of Liu *et al.* [224].

Pairs of corresponding AA and CG snapshots are generated by mapping AA configurations onto the CG resolution. Three data sets are constructed from uncorrelated snapshots selected from different trajectories simulated at $T = 313$ K, 453 K, and 568 K. To cover a wide range of conformational space, each atomistic simulation was initialized from a different structure: The simulation at 313 K started from a β structure, at 453 K from an α structure and at 568 K from an amorphous configuration. The system includes 36 polystyrene chains and each chain consists of 10 monomers. While only 12 snapshots are used for training, further 78 snapshots are used for testing.

The fine-to-coarse mapping is based on the CG model developed by Fritz *et al.* [227]. Each monomer is mapped onto two beads of different types, denoted A for the chain backbone and B for the phenyl ring (see Fig. 5.1). Bonds are formed only between backbone and phenyl ring beads, i.e. the CG polymer is represented as a linear chain A-B-A-$B \cdots$. The close connection between backbone beads A is reproduced indirectly by angular potentials. While the CG model is parameterized in the melt, Liu *et al.* have shown that it is transferable to the crystalline phase, where it stabilizes the experimentally observed α and β polymorphs [224].

5.1.2 Octane and Cumene

Octane and cumene are small hydrocarbons. While octane is an acyclic alkane, cumene is aromatic. MD simulations of octane and cumene liquids are performed using the molecular dynamics package GROMACS 5.0 [226]. The GROMOS force field is used and topologies are generated by AUTOMATED TOPOLOGY BUILDER [228]. Note that the GROMOS and sPS force fields differ in parameterization strategies. While both force fields aim at reproducing thermodynamic properties, the GROMOS force field is designed for a broad range of molecular systems, while the

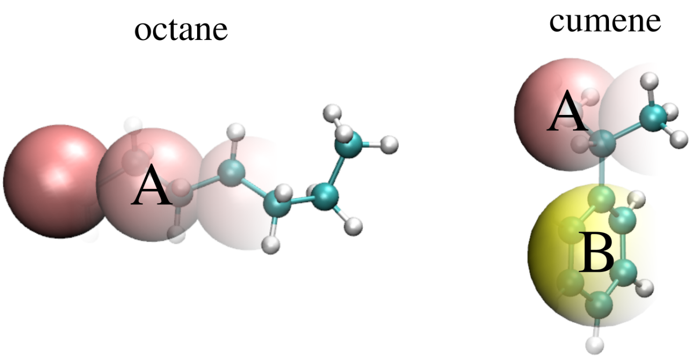

FIGURE 5.2: CG and AA representation of octane and cumene. The CG mapping is based on the mapping for sPS.

parameterization for sPS is custom-built from a specific force field designed for benzene. This leads to evident differences in force field parameters, especially in terms of the non-bonded Lennard-Jones interactions and partial charges.

The simulation boxes of octane and cumene contain 215 and 265 molecules, respectively. MD simulations are performed in the NPT ensemble using the velocity rescaling thermostat and the Parrinello–Rahman barostat. The integration time step is set to 1 fs and both systems are sampled at 350 K.

As illustrated in Fig. 5.2, the fine-to-coarse mapping is based on the mapping for sPS. Cumene is mapped onto one bead of type B for the phenyl ring and two beads of type A for the backbone, each containing a methyl group and sharing the CH group connected to the phenyl ring. Octane is mapped onto four beads of type A, where neighboring A beads share a CH_2 group.

5.1.3 Baseline Method

The results of DBM are compared to a generic backmapping strategy, as described in Sec. 2.4.2. Specifically, the backmapping script developed by Wassenaar *et al.* is utilized [47]. In a first step, this method places each particle on the weighted average position of the CG beads it corresponds to and optionally adds a random displacement. In addition, the protocol allows the user to apply geometric modifiers setting the alignment of the next particle "cis", "trans", "out", or "chiral" with respect to the other particles. The modifiers are crucial for the performance of this method and require a careful adjustment by the user.

After the initial structure is generated, the protocol by Waasenaar *et al.* continues with multiple cycles of force field based energy minimization for relaxation. Here, the first cycle consists of 200 steps and takes only bonded interactions into account. Afterwards, all interaction potentials are turned on and energy minimization continues with a total number of 5000 steps. The original protocol continues with several cycles of position restrained MD simulations to equilibrate the relaxed system.

However, comparing DBM with such equilibrated backmapped structures is not insightful, since applying MD simulations would evidently reproduce the Boltzmann distribution, which is already captured by the reference test set. In order to highlight the capability of DBM to generate equilibrated molecular structures without MD, the script by Waasenaar *et al.* is stopped after the relaxation, such that reconstructions generated by DBM can be compared to energy-minimized configurations prior to MD.

5.1.4 Specifications of DBM

DBM deploys a convolutional neural network (CNN) architecture with residual connections for the generator g and critic c [229]. A detailed description of the network architecture can be found in Fig. A.1 of the appendix. Training is performed using the Adam optimizer [230]. The cutoff distance applied for the local environments is set to $r_{\mathrm{cut}} = 1.2$ nm. To prevent numerical instabilities in the beginning of the training, the prefactor for the regularization term based on the potential energy is set initially to $\lambda_{\mathrm{pot}} = 0$ and increased smoothly to $\lambda_{\mathrm{pot}} = 0.01$. The prefactor scaling the weight of the gradient penalty term is set to $\lambda_{\mathrm{gp}} = 0.1$ throughout the training. To obtain reliable gradients for the generator g, the critic c is trained five times in each iteration while the generator g is trained once.

The autoregressive approach of DBM is prone to accumulate errors, i.e. misplaced atoms can hinder g to find suitable positions for subsequent atoms. As a remedy, the potential energy is used during inference in order to spot and reject outliers. For each local environment, a mini-batch is constructed by random rotations around the director axis (see Sec. 4.3.1). Since the CNN architecture is not rotational equivariant, predictions will slightly differ depending on the relative orientations. In addition, different prior samples $\mathbf{z}$ are used for each element in the mini-batch to further increase variations of generated structures. As a straightforward solution to mitigate the effect of misplaced atoms, the structure with the lowest potential energy is selected from the generated ensemble. While this simple procedure performs well in practice, it should be noted that it might introduce a bias towards low-energy structures. In addition, hydrogens are removed from the current and adjacent beads for the reconstruction of heavy atoms, such that misplaced hydrogens do not affect the positioning of heavy atoms.

5.2 General Performance

In this section, the general performance of DBM is probed. To this end, DBM is trained on the high-temperature data set at 568 K, where the sPS system is in a melt state. After training, the model is applied to test data, i.e. hold-out data at the same temperature.

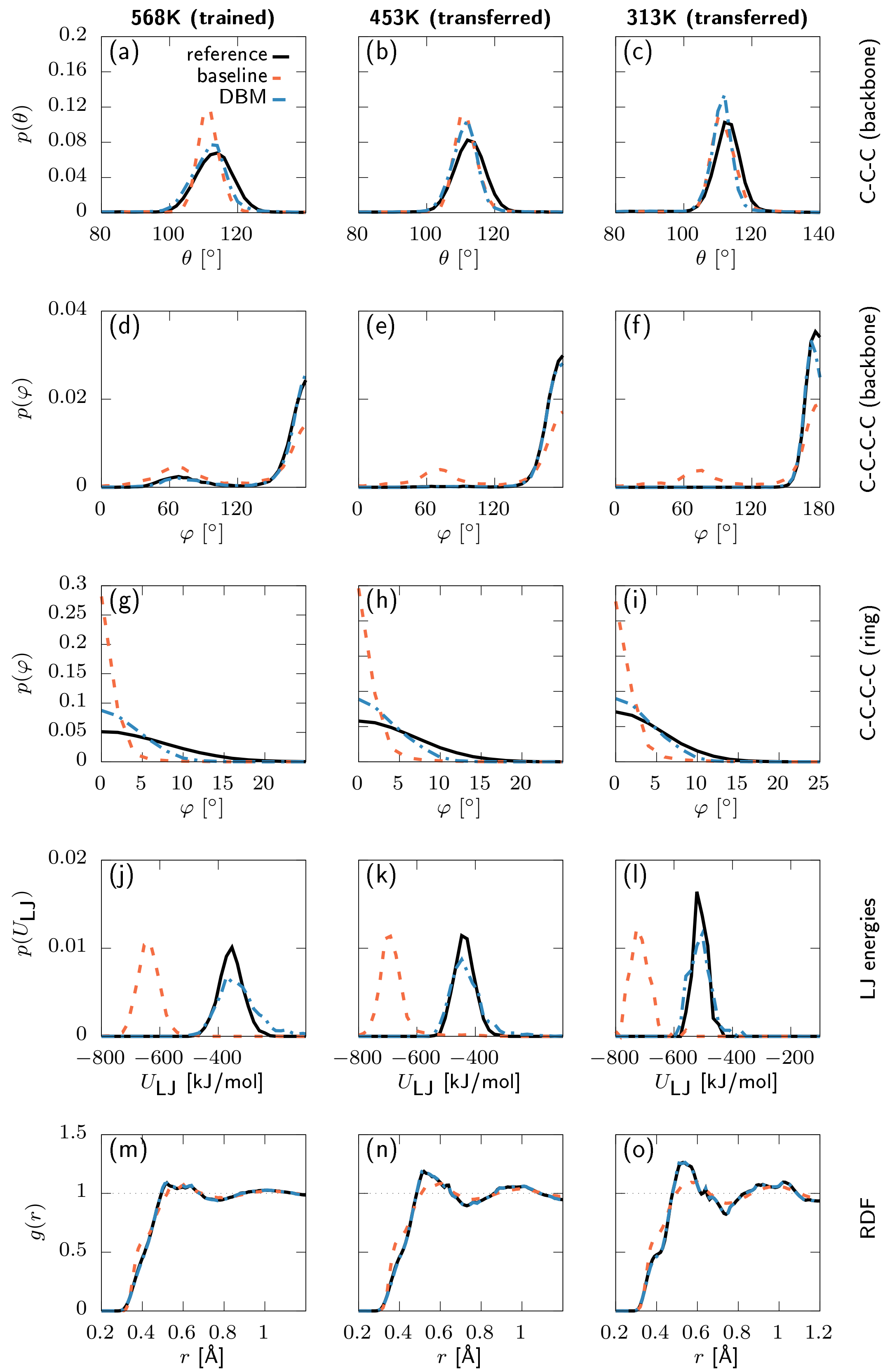

FIGURE 5.3: Canonical distributions for various force field interaction terms at (left) $T = 568$ K, (middle) $T = 453$ K and (right) $T = 313$ K for reference structures (black), structures generated with a baseline method based on energy-minimization (red), and the new method DBM (blue). The ML model DBM is trained solely on the high-temperature data (left), but transferred to lower temperatures (middle and right). (a)–(c) C-C-C backbone angle, (d)–(f) C-C-C-C backbone dihedral, (g)–(i) C-C-C-C improper dihedral, (j)–(l) Lennard-Jones energies, and (m)–(o) radial distribution functions, g(r), of the non-bonded carbon atoms. Reprinted from [208].

Fig. 5.3 displays distribution functions for several structural and energetic properties of sPS. The distributions of intramolecular carbon backbone angle and dihedral, shown in panels (a) and (d), are in excellent agreement with the reference distributions. On the other hand, structures generated with the baseline method display distributions that are more compressed compared to the reference system, which is expectable from an approach based on energy minimization. As shown in panel (g), the distribution for the carbon improper dihedral of the phenyl group is slightly biased towards smaller angles for configurations generated with DBM. However, the small range of angles, imposed by the planarity of the ring, has to be emphasized. The distribution of the baseline method is even more peaked, i.e. fluctuations around the planar structure are significantly suppressed.

A very important aspect towards generating well-equilibrated configurations in a condensed-phase environment is the correct reproduction of Lennard-Jones energies. Panel (j) displays the distribution of Lennard-Jones energies obtained separately for each chain. While structures generated with DBM show slightly too large high-energy tails, the overall match with the reference distribution is remarkably good. The baseline method systematically and drastically over-stabilizes the system.

Further, the pair correlation function $g(r)$ for non-bonded carbon pairs is analyzed in panel (m). Structures generated with DBM show an excellent agreement with the reference distribution indicating an accurate reconstruction of pairwise distances. The baseline method is also able to generate pair correlations with high accuracy, but still displays some discrepancies to the reference system.

5.3 Temperature Transferability: From Melt to Crystal

After successfully recovering the state point DBM was trained on, the ability of the model to transfer across temperatures is probed. As illustrated in Fig. 5.4, the training of DBM is fixed to the high-temperature ensemble, while testing is performed at lower temperatures *without* reparameterization. Specifically, the model is trained at 568 K and tested at 453 K and 313 K. Note that the sPS system undergoes a phase transition at $\approx$ 450 K, going from a melt to a crystalline state with different polymorphs. In particular, the test data set contains snapshots of the α and β polymorphs that differ in the packing of the sPS chains.

The transferability of DBM to the crystalline phase of sPS is analyzed in terms of structural and energetic distributions. In addition, a higher-order investigation facilitated by the Sketch-map (SM) algorithm is performed to obtain a two-dimensional projection of configuration space [160, 161]. Finally, AA MD simulations initialized from reference and backmapped structures are evaluated.

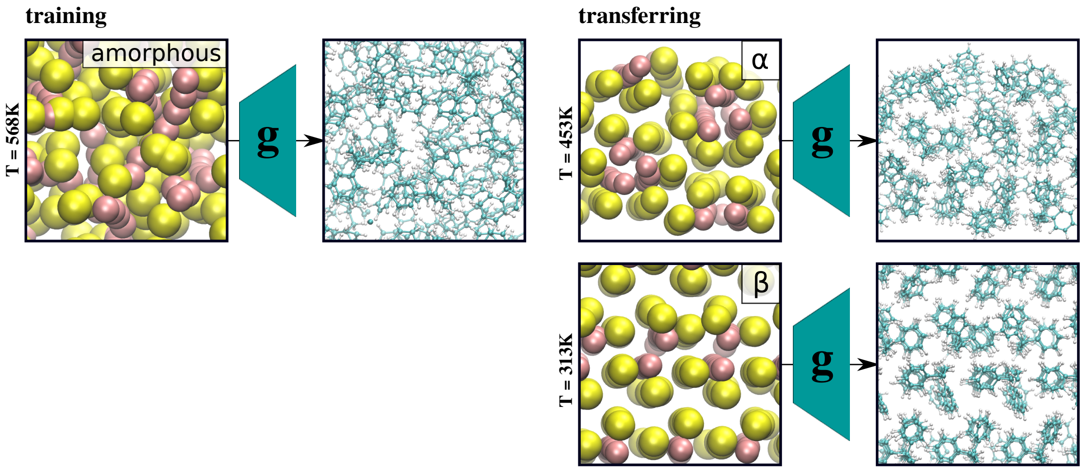

FIGURE 5.4: Polymorphism of Polystyrene. At a high temperature ($T = 568$ K), the polymeric system is in a melt state. At lower temperatures, the CG model mostly stabilizes the α polymorph at $T = 453$ K and the β polymorph at $T = 313$ K. DBM is trained solely on the high-temperature ensemble and its transferability to the lower temperatures is probed. Reprinted from [208].

5.3.1 Distributions of Structural and Energetic Features

Distributions of structural and energetic properties at 453 K and 313 K can be found in the middle and right column of Fig. 5.3, respectively. The reference system displays a number of significant changes upon cooling: distributions of angles become more compressed, the side peak in the backbone dihedral vanishes, the distributions of Lennard-Jones energies are shifted towards lower energies and the pair correlation function of non-bonded carbon atoms is more peaked.

The ML model adapts remarkably well to the crystalline phase in terms of the angle and dihedral distributions shown in panels (b,c,e,f,h,i): DBM yields distributions that follow the reference distributions and become more compressed upon cooling. Lennard-Jones energies displayed in panels (k) and (l) are also shifted and match with the reference distributions. Moreover, non-bonded pair correlations in the crystalline phase are reproduced with remarkable accuracy, as indicated in panels (n) and (o).

On the other hand, the baseline method does not adapt well to lower temperatures. Due to the energy minimization, similar distributions are obtained as for the high-temperature data, which becomes especially apparent for the side peak of the backbone dihedral in panels (e) and (f), as well as the flat pair correlation function in panels (n) and (o).

5.3.2 Sketch-map

Evaluating large-scale structural features beyond pair-statistics is challenging, since the high dimensionality of the system does not allow for a direct visualization of the configuration space. For this reason, dimensionality reduction is applied to further

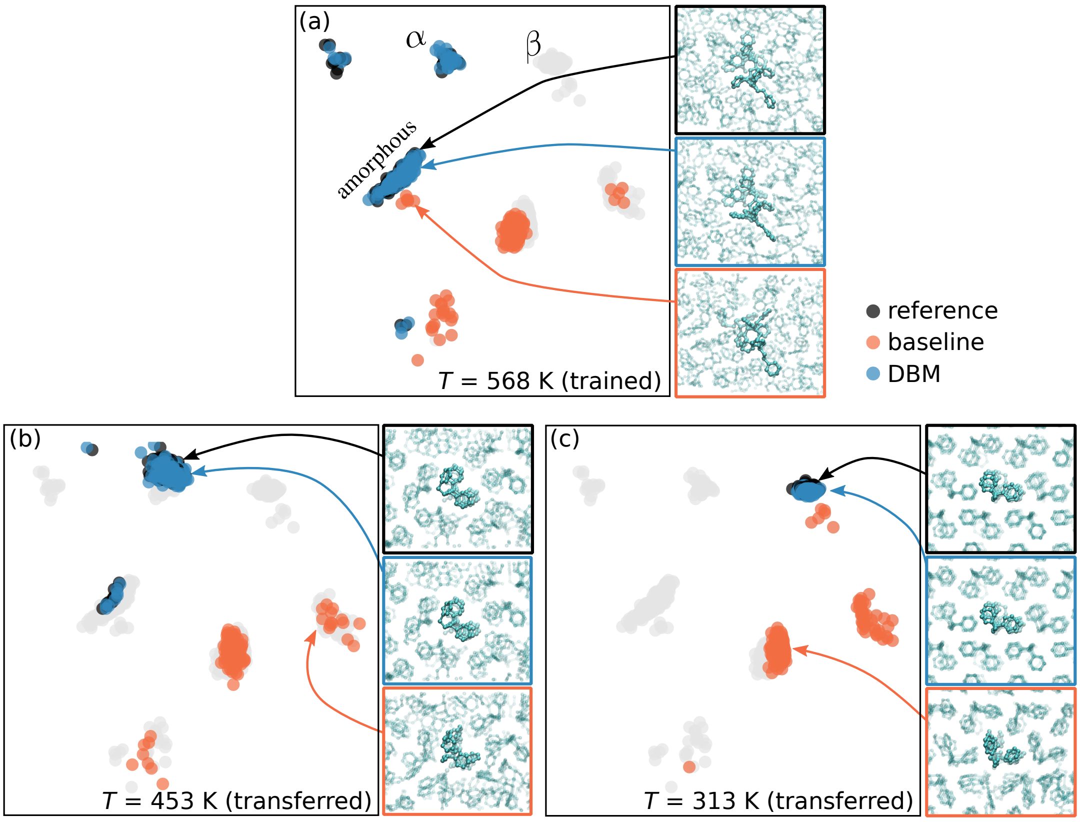

FIGURE 5.5: Two-dimensional projection of the configuration space obtained with SM. Each point represents a sPS chain in a condensed-phase environment at (a) $T = 568$ K, (b) $T = 453$ K and (c) $T = 313$ K. Projections for reference structures (black), backmapped structures obtained with the baseline method (red) and with DBM (blue) are shown. For each panel, snapshots are backmapped from identical CG configurations. A projection of the entire data set including all temperatures is displayed in gray for visual guidance. Reprinted from [208].

examine the model's accuracy at higher order. As explained in Sec. 3.2.3, linear dimensionality reduction techniques are often insufficient to capture the structure of data obtained from MD trajectories. Therefore, the non-linear SM algorithm is applied to build a two-dimensional map representing proximity relationships between sPS chains [160, 161].

The descriptors for the sPS chains consist of a set of representations for the local environments $\mathcal{H}$ centered around alternating backbone carbon atoms that are directly linked to a phenyl group. The pairwise distance between two such environments is encoded using a similarity kernel $k(\mathcal{H}, \mathcal{H}') = \mathbf{p}(\mathcal{H})\mathbf{p}(\mathcal{H}')$ based on the normalized many-body smooth overlap of atomic position (SOAP) representation $\mathbf{p}(\mathcal{H})$ [231]. Hydrogen atoms are neglected in the SOAP representation. To compare two sPS chains a and b, the covariance matrix

$$C_{ij}(a,b) = \mathbf{p}(\mathcal{H}_i^a)\mathbf{p}(\mathcal{H}_j^b) \tag{5.1}$$

is computed, which contains the complete information of the pairwise similarity of all local environments that are taken into account between the two structures. In order to obtain a global similarity kernel $k(a, b)$, the covariance matrix $C_{ij}(a, b)$ has to be mapped to a single scalar value, which is achieved using a regularized entropy match kernel [232].

Fig. 5.5 displays the two-dimensional projection obtained with SM, where each point represents a single sPS polymer chain and its local environment. The projection of the reference data yields a number of distinct clusters: The low-temperature data at 313 K forms a single cluster (panel (c)), which can be associated with the β polymorph. The high-temperature data at 568 K (panel (a)) is mapped to multiple clusters indicating more diversity, i.e. it includes amorphous, α and other structures. The data set at an intermediate temperature of 453 K (panel (b)) is mapped mostly to the cluster corresponding to the α polymorph, but also contains some amorphous and other structures.

Structures obtained with DBM display a significant overlap with the reference points for all three data sets indicating closeness in configuration space and a high structural fidelity of the backmapped structures. This is in strong contrast to the energy-minimized structures obtained with the baseline method, which map to different areas in the two-dimensional projection of configuration space compared to the reference configurations.

5.3.3 MD Simulation

Backmapped structures that serve as a starting point for further MD simulations typically require lengthy preparations, such as energy minimization, temperature ramp up phase and thermostat/barostat equilibration. In the following, the high quality of backmapped structures obtained with DBM is demonstrated by running MD simulations *without* any heat-up.

The simulations are carried out in the NPT ensemble using the velocity rescaling thermostat and the Parrinello-Rahman barostat. Initial velocities are generated according to a Maxwell distribution and an integration timestep of 1 fs is used.

Fig. 5.6 displays the time evolution of the potential energy during simulations at (a) $T = 313$ K and (b) $T = 568$ K. Simulations starting from reference or backmapped structures obtained with DBM show a similar evolution of the potential energy at both temperatures and reach a steady value after ≈ 100 ps. On the other hand, simulations starting from backmapped structures obtained with the baseline method display a different behavior: The potential energy of simulations performed at 313 K settles at significantly higher energies compared to simulations starting from reference or DBM structures. This indicates poorly initialized structures that get trapped into local minima with high energy barriers. However, this issue is not apparent at 568 K, where all simulations display a similar behavior independent of their initialization. This can be rationalized with the higher temperature that increases the probability of escaping local minima.

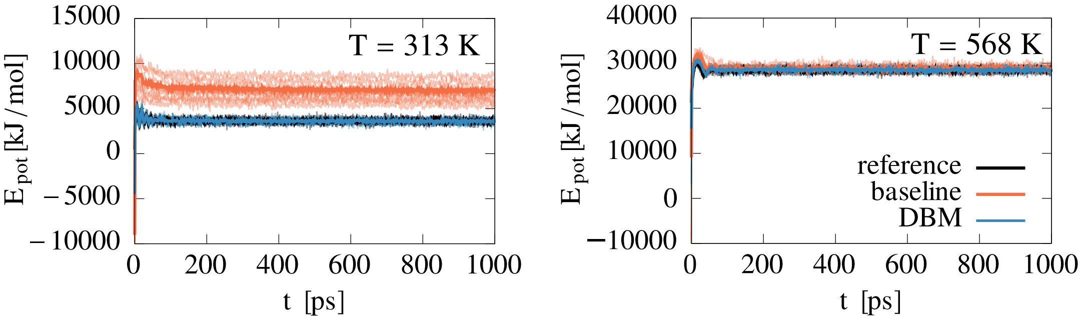

FIGURE 5.6: Evolution of the potential energy in MD simulations without heat-up starting from reference structures (black), backmapped structures obtained with the baseline method (red) and with DBM (blue). Reprinted from [208].

5.4 Chemical Transferability: From Small Molecules to Polymers

In this section, the transferability of DBM across chemical space is explored. In particular, the generalization of the model beyond the chemistry used for training is probed by recycling the learned local correlations to make predictions for molecules absent from the training data set. As illustrated in Fig. 5.7, the model is trained on molecular liquids of octane and cumene molecules. After training, the model is reused for the more complex polymeric system consisting of sPS molecules. While sPS shares some features with cumene and octane, it is still sufficiently complex to study the limitations of the generalization. As such, the pertinent but imperfect match between the small molecules and sPS offers a stringent backmapping exercise. In the following, the performance of chemically-transferred models is compared to the performance of chemically-specific models, i.e. models trained directly on sPS. In addition, the role of the different types of force field based regularization, introduced in Sec. 4.5, is explored by comparing their impact on the performance of the model, especially regarding chemical transferability. In particular, three different regularization configurations are applied for the training of the model: Either $\mathcal{C}_1$ ("energy minimizing") or $\mathcal{C}_2$ ("energy matching") terms are added to the cost-function of the generator or no regularization is used.

The training set consists of 15 snapshots of the octane system and 8 snapshots of the cumene system in the liquid state. After training, DBM is applied to a test set consisting of 20 snapshots of the sPS melt. While octane and cumene liquids are simulated at $T = 350$ K, the sPS melt is simulated at $T = 568$ K. The discrepancy in the temperature between the training and test sets is a consequence of the different boiling and melting points of the molecules, as the model's transferability shall be probed in the liquid/melt state. However, as shown in Sec. 5.3, the learned local correlations are weakly sensitive to changes in the temperature.

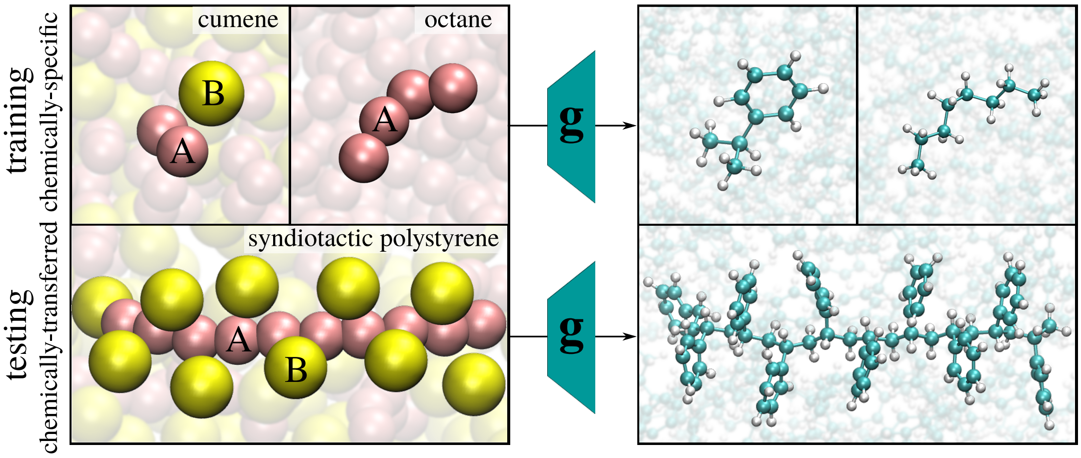

FIGURE 5.7: AA and CG representations of different molecules. A similar fine-to-coarse mapping is applied for all molecules, i.e. *A* beads represent the chain backbone and *B* beads represent phenyl rings. In order to probe the chemical transferability of DBM, it is trained solely on octane and cumene liquids and then applied to the more challenging polymeric system of sPS. Reprinted from [223].

5.4.1 Distributions of Structural and Energetic Features

Figs. 5.8-5.10 display various distribution functions for structural and energetic properties of sPS derived for reference structures and structures generated with DBM. Results are shown for chemically-specific models (left), i.e. models trained directly on sPS, and chemically-transferred models (right), i.e. models trained solely on octane and cumene configurations.

Angle distributions can be found in Fig. 5.8. The general performance of chemically-transferred models varies and deviates from the chemically-specific models. The largest discrepancy can be found for the carbon backbone angle displayed in panels (a) and (b). While models trained directly on sPS reproduce the angles of the carbon chain with remarkable accuracy, models trained on cumene and octane generate structures with overly broad distributions. However, the overall accuracy of further angles (panels (c) - (h)) reproduced by the chemically-transferred models is exceedingly satisfactory. The role of the regularization applied during training is not significant.

Various dihedral distributions are displayed in Fig. 5.9. Again, the accuracy of chemically-transferred models varies compared to chemically-specific models. All models are able to reproduce the planarity of the phenyl ring with high accuracy, as displayed in panels (c)-(d) and (g)-(h). While the distributions for the improper dihedrals are slightly too compressed compared to the reference, the small range of the distributions has to be emphasized. However, models trained directly on sPS outperform the chemically-transferred models in terms of the accuracy of the backbone dihedrals, as shown in panels (a)-(b) and (e)-(f). In particular, the chemically-transferred models fail to reproduce the height of the main peak and are not able to reproduce the side peak of the proper backbone dihedral. The configuration of the regularization again has no impact on the observed distributions.

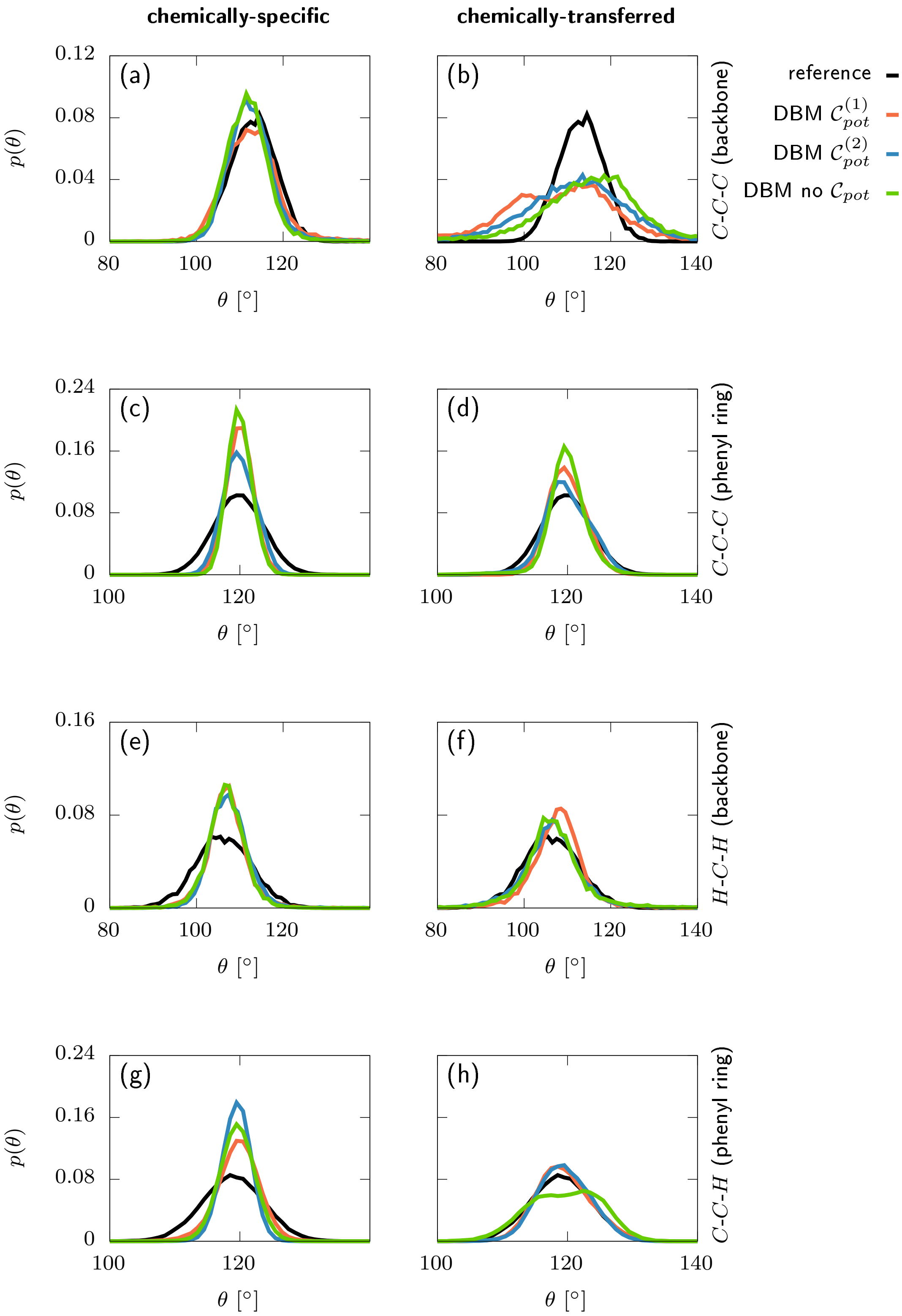

FIGURE 5.8: Canonical distributions for sPS at $T = 568$ K. (a)-(h) Various angle terms for reference and backmapped structures. Backmapping is performed with DBM using different regularization terms during training. Left: Chemically-specific models trained on sPS melts at $T = 568$ K. Right: Chemically-transferred models trained on octane and cumene liquids at $T = 350$ K. Reprinted from [223].

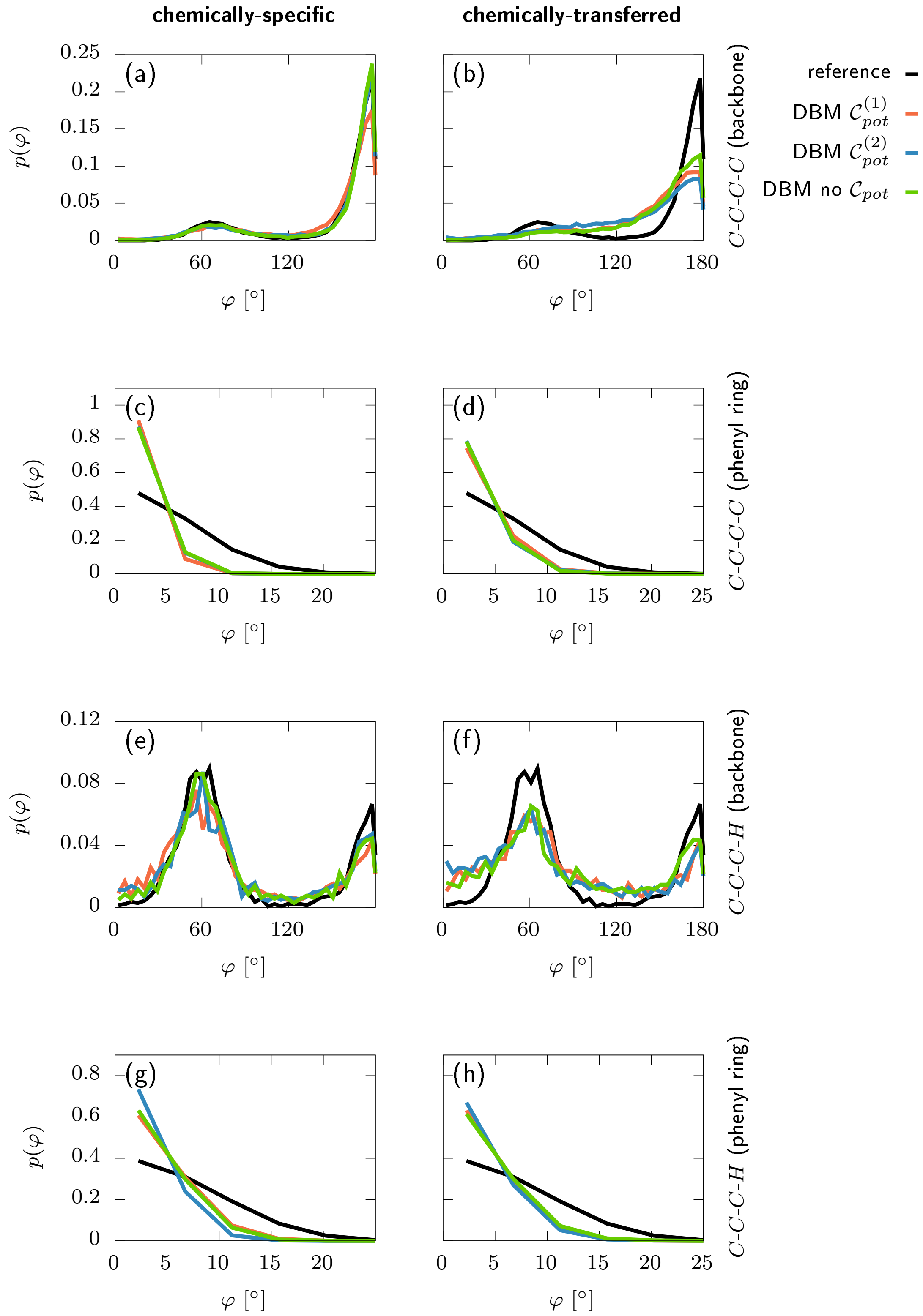

FIGURE 5.9: Canonical distributions for sPS at $T = 568$ K. Various dihedral terms for reference and backmapped structures. Backmapping is performed with DBM using different regularization terms during training. (a), (b), (e), and (f) Proper dihedral; (c), (d), (g), and (h) improper dihedral. Left: Chemically specific models trained on sPS melts at $T = 568$ K. Right: Chemically transferred models trained on octane and cumene liquids at $T = 350$ K. Reprinted from [223].

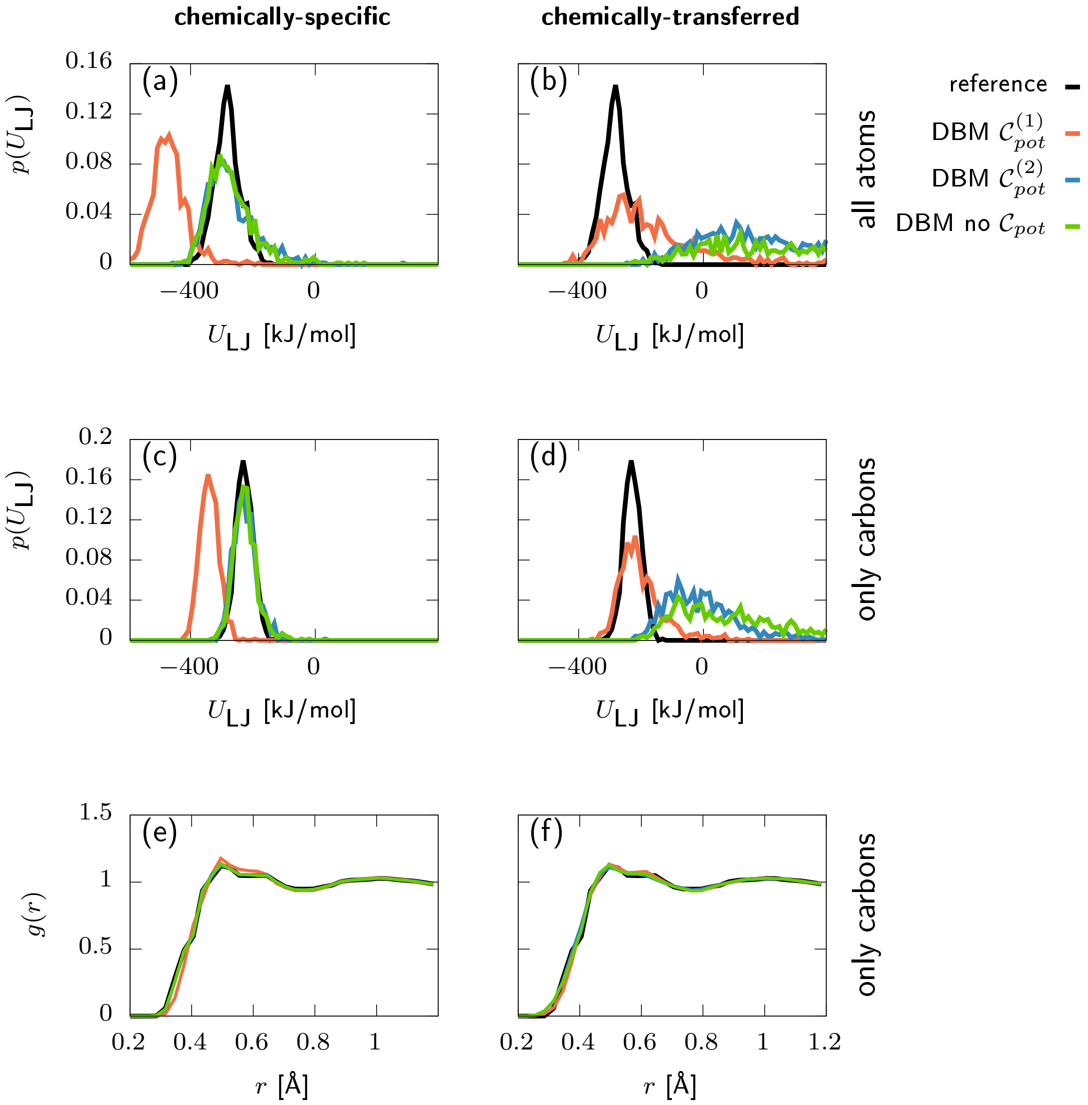

FIGURE 5.10: Canonical distributions for sPS at $T = 568$ K. (a) and (b) Lennard-Jones energies for all atoms, (c) and (d) Lennard-Jones energies for carbon atoms, (e) and (f) radial distribution functions, $g(r)$, of the non-bonded carbon atoms. Distributions for reference and backmapped structures are shown. Backmapping is performed with DBM using different regularization terms during training. Left: Chemically-specific models trained on sPS melts at $T = 568$ K. Right: Chemically-transferred models trained on octane and cumene liquids at $T = 350$ K. Reprinted from [223].

The ramification of the applied regularization during training becomes most evident in the distributions of the Lennard-Jones energies obtained for each sPS chain separately, which are displayed in Fig. 5.10. Chemically-specific models trained with regularization $\mathcal{C}_{\text{pot}}^{(2)}$ or without regularization reproduce the reference Lennard-Jones energies with high accuracy. Carbon-only Lennard-Jones energies (panels (c) and (d)) match the reference distribution almost perfectly. Lennard-Jones energies taking hydrogens into account (panels (a) and (b)) display slightly too large high-energy tails for the backmapped structures. In contrast, applying regularization $\mathcal{C}_{\text{pot}}^{(1)}$ over-stabilizes the system and yields a significant shift of the distribution towards lower energies. However, these observations turn around for the chemically-transferred models: Here, regularization $\mathcal{C}_{\text{pot}}^{(1)}$ improves the performance dramatically compared to models trained with $\mathcal{C}_{\text{pot}}^{(2)}$ or without regularization. While the chemically-transferred model trained with $\mathcal{C}_{\text{pot}}^{(1)}$ reproduces the Lennard-Jones energies remarkably well, except for a small tail towards high energies, the other models yield backmapped structures with systematically too large Lennard-Jones energies.

The pair correlation function obtained for pairs of non-bonded carbon atoms is shown in Fig. 5.10 (e)-(f). All models are able to reproduce the pair correlation with high accuracy.

5.4.2 Sketch-map

Similarly to the analysis in the previous section, the SM algorithm is used to probe the accuracy of backmapped structures at higher order. In particular, SM is applied to project environments of sPS monomers onto a two-dimensional embedding. To this end, local environments of monomers are centered at backbone carbons that are connected with a phenyl group and the similarity between two environments is computed based on the SOAP representation.

Fig. 5.11 (a) displays the obtained embedding for reference structures. The local environments of 720 sPS monomers are used to infer landmarks for the two-dimensional map. Afterwards, further 1440 local environments are projected onto the SM space guided by the landmarks. Fig. 5.11 (b) shows the projections of backmapped structures obtained with a chemically-specific model and a chemically-transferred model. The underlying CG structures correspond to the atomistic structures used for the projections of reference structures in Fig. 5.11 (a). Both models are trained with $\mathcal{C}_{\text{pot}}^{(1)}$. Further plots for models trained with $\mathcal{C}_{\text{pot}}^{(2)}$ or without regularization can be found in Fig. A.2 in the appendix. The high structural fidelity of both, the chemically-specific and the chemically-transferred model, is highlighted by the strong overlap of the projections obtained for reference and backmapped structures.

The two dimensional representations obtained with the SM algorithm form a number of distinct clusters. As such, points assigned to the same cluster indicate

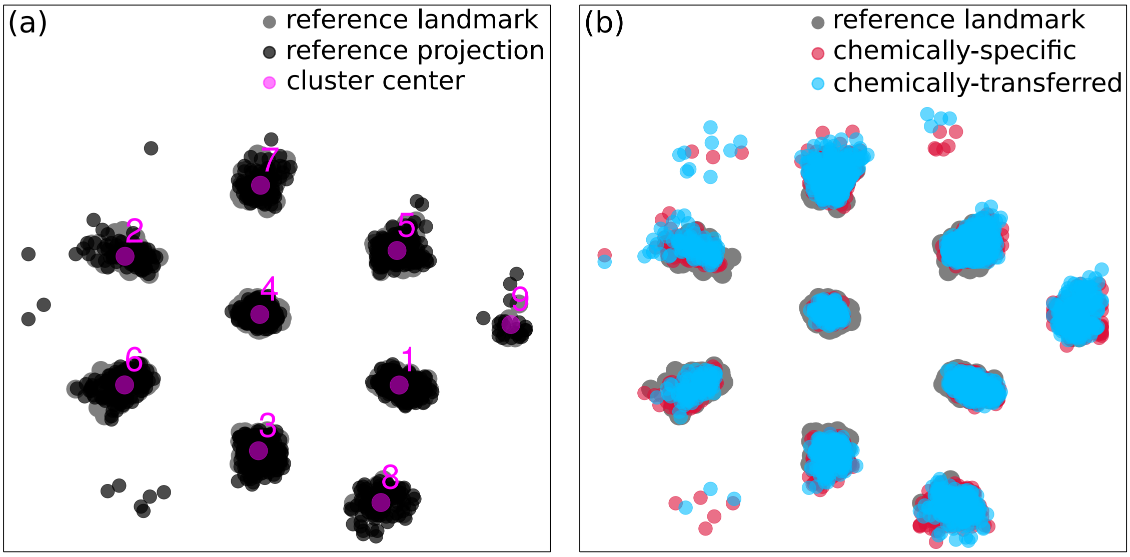

FIGURE 5.11: Low-dimensional representation of the local environments of sPS monomers at $T = 568$ K. For each panel, snapshots are backmapped from identical CG configurations. (a) Landmarks (gray) and projections (black) of reference structures and the obtained cluster centers. (b) Landmarks of reference structures (gray) and projections of structures generated with chemically-specific (red) and chemically-transferred (blue) models trained with $\mathcal{C}^{(1)}_{\text{pot}}$. Reprinted from [223].

closeness in conformational space. For a further analysis, cluster centers are identified using the k-means algorithm. Each monomer embedding is assigned to the closest cluster. This yields a confusion matrix that enables to compare the cluster assignment of reference and backmapped structures. While the diagonal elements of the confusion matrix hereby refer to reference and backmapped structures that get mapped onto the same cluster, off-diagonal elements indicate a change of the cluster assignment upon coarse-graining and backmapping. The results for chemically-specific and chemically-transferred models trained with different regularization configurations can be found in Fig. 5.12. Interestingly, the confusion matrix becomes most diagonal for models trained without regularization displayed in panels (c) and (g), respectively. However, the reduced resolution of the CG conformational space implies that an ensemble of microstates is associated with a single CG structure, as described in Sec. 2.4. The ensemble of microstates associated with the same CG structure might span a broad region in conformational space, which is not guaranteed to map onto the same cluster. As such, the diagonality of the confusion matrix is not necessarily a proper indicator for the quality of the backmapped distribution. More importantly, the relative populations of the clusters have to be reproduced, as this implies an accurate coverage of conformational space. Results comparing the relative cluster populations can be found below each confusion matrix in Fig. 5.12. Chemically-specific models trained with $\mathcal{C}^{(2)}_{\text{pot}}$ or without regularization yield an excellent match of the relative populations. On the other hand, all chemically-transferred models display a similar accuracy independent of the regularization and yield relative populations that differ from the reference system.

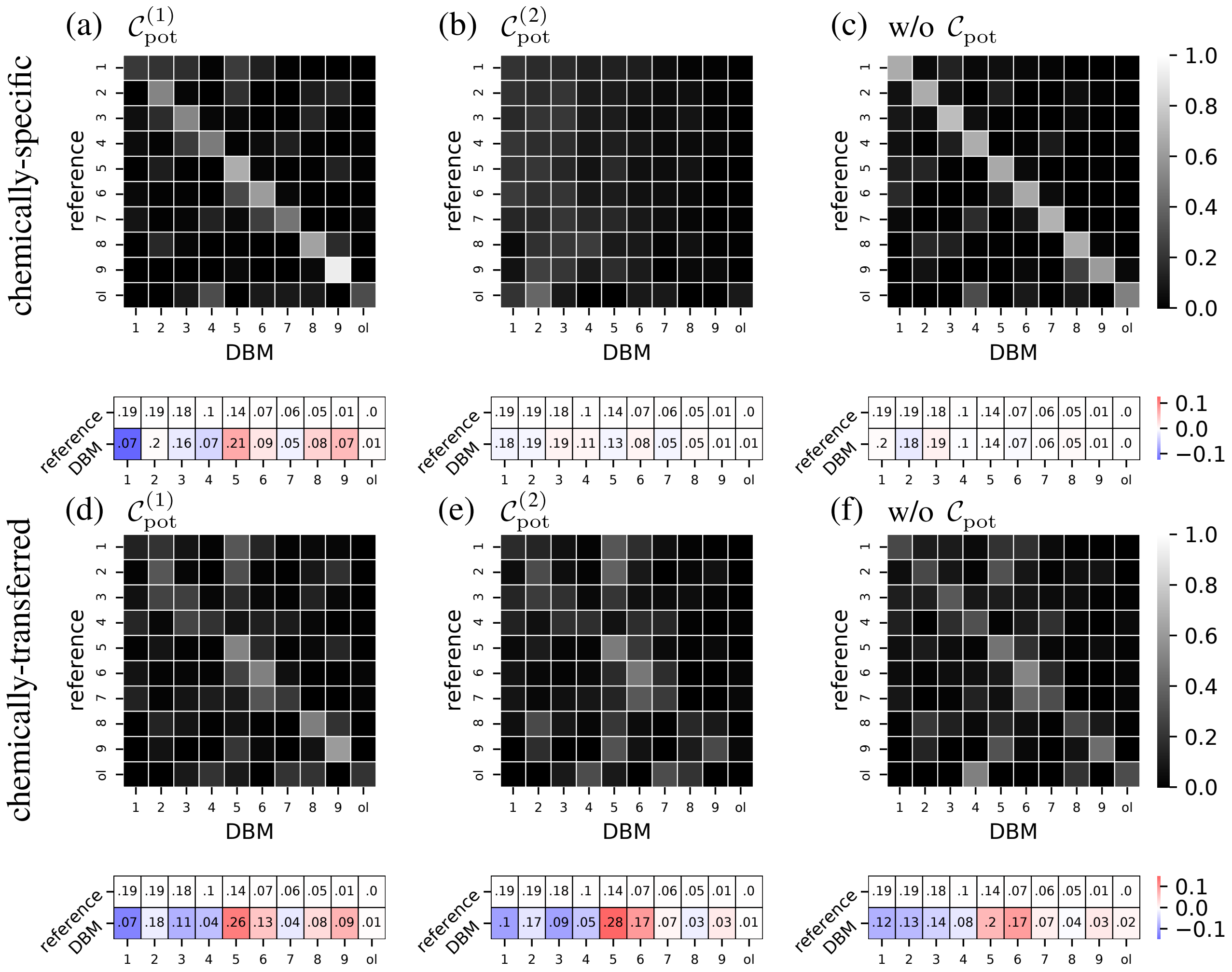

FIGURE 5.12: (Top) Confusion matrix for the different clusters obtained in the two-dimensional SM. (Bottom) Relative populations of the clusters. (a)–(c) chemically-specific models trained with (a) $\mathcal{C}_{\text{pot}}^{(1)}$, (b) $\mathcal{C}_{\text{pot}}^{(2)}$, and (c) no regularization. (d)–(f) chemically-transferred models trained with (d) $\mathcal{C}_{\text{pot}}^{(1)}$, (e) $\mathcal{C}_{\text{pot}}^{(2)}$, and (f) no regularization. "ol" denotes outlier. Reprinted from [223].

5.5 Discussion

This chapter evaluates the performance and transferability of the ML-based backmapping scheme DBM, which is introduced in Chpt. 4. In this section, the main results are summarized and discussed.

- **General reverse-mapping capability of DBM:** The ability of DBM to reproduce a reference AA distribution from CG configurations is probed. To this end, DBM is applied to high-temperature data of a condensed-phase molecular system of sPS chains. Based on an evaluation of structural and energetic distributions, it is found that the ML-based method yields well-equilibrated configurations for this particular state point. In addition, a baseline method based on geometric rules and energy minimization is applied, which overstabilizes the system and therefore does not reproduce the specific state point accurately.

- **Transferability across different state points:** To probe the temperature transferability, the training of DBM is fixed to melt configurations obtained at a high temperature. Afterwards, the model is transferred to crystalline structures at lower temperatures. DBM retains the excellent performance it has shown for the high-temperature state point and reproduces structural and energetic distributions of the reference system with remarkable accuracy. A higher-order investigation, facilitated by the SM algorithm, highlights the structural fidelity. Moreover, MD simulations initialized from backmapped structures display a similar behavior as simulations initialized from reference structures. In summary, the local correlations learned in the melt transfer remarkably well to the crystalline state point. On the other hand, the baseline method displays limited transferability to the crystalline phase. In particular, MD simulations starting from backmapped configurations in the crystalline phase get stuck in local minima. As such, human intervention would be required to achieve proper equilibration, which hinders the automation of such reverse-mapping processes.

  The remarkable temperature transferability of DBM can be rationalized in terms of a scale-separation: The model learns to reproduce well-equilibrated local correlations while large-scale features are dictated by the CG configuration. As such, the backmapped structure is composed of two sources of information, 1) the learned local features and 2) the CG structure. It can be hypothesized that most of the temperature dependence is carried by the CG structure, as shown by Liu *et al.* that the applied CG model reproduces the crystallization transition remarkably well [224]. Local features, on the other hand, are less temperature sensitive, since they correspond primarily to covalent interactions that operate on energy scales significantly larger than $k_B T$. As such, local correlations learned in the melt are transferable to the crystalline phase. However, it is not clear whether the other direction, i.e. training at a low temperature, where the system is in the crystalline phase, should lead to satisfying transferability at high temperatures, given the broader conformational space spanned.

- **Transferability across chemical space:** To probe the chemical transferability, the training of DBM is fixed to liquids of octane and cumene. Afterwards, the model is transferred to the more complex sPS system without retraining. The performance of such chemically-transferred models varies in terms of bonded interactions: While the learned local correlations from octane and cumene allow for an accurate reconstruction of phenyl groups, reconstructed polymer backbones display discrepancies compared to the reference system. On the other hand, chemically-transferred models retain their capability to reproduce non-bonded features in the challenging condensed-phase environment. They are able to recover the distribution of Lennard-Jones energies with remarkable accuracy and match the pair correlation function of the reference distribution virtually identically. The high structural fidelity of backmapped structures is further highlighted by a higher-order investigation facilitated by the SM algorithm. Although backmapped structures and their reference counterparts are not necessarily mapped onto the same cluster, as indicated by the confusion matrix, the correct spots in the two-dimensional projection of conformational space are covered. However, discrepancies in the relative statistical weights of reference and backmapped microstates are observed.

  The overall encouraging performance of chemically-transferred models demonstrates that small-scale features can be shared between different molecules. In other words, it highlights the capability of DBM to interpolate across parts of chemical space due to its local environment representations. Specifically, the local correlations learned from octane and cumene liquids transfer to a great extend to sPS melts. However, the limits of generalization are shown as well, indicated by the limited quality of the reconstructed carbon backbone. It can be hypothesized that accuracy bottlenecks arise from missing features. In particular, local environments of backbone carbons connecting monomers are absent in the training examples. As such, training on an increasing number of building blocks should systematically improve the transferability of the backmapping model. Another important aspect affecting the transferability of DBM are force field inconsistencies between the molecules. Consequently, conformational spaces are evidently incoherent and features found for fragments of sPS and cumene/octane are more dissimilar than first expected.

  Finally, the effect of the different regularization terms applied during training is investigated. The applied regularization has marginal impact on the distributions associated with covalent interactions. However, distributions for the non-bonded Lennard-Jones energies are significantly affected by the setting of the regularization. This can be rationalized taking the functional form of the interactions the regularization terms are based on into account: Harmonic or periodic potentials are applied for the bonded interactions, which react moderately to shifts of the atomic positions. On the other hand, the Lennard-Jones potential is more sensitive, i.e. small shifts of atomic positions can yield a dramatic change of the energy by several orders of magnitude. As such, gradients computed for energy-based regularization terms are dominated by the Lennard-Jones contributions. However, the energy-matching regularization $\mathcal{C}_{\text{pot}}^{(2)}$ has an overall minor impact compared to training without regularization. On the other hand, application of the energy-minimizing regularization $\mathcal{C}_{\text{pot}}^{(1)}$ improves the performance of chemically-transferred models dramatically and yields Lennard-Jones distributions that match the reference distribution remarkably well. However, application of $\mathcal{C}_{\text{pot}}^{(1)}$ for chemically-specific models over-stabilizes the system and yields structures with too low energies. It can be hypothesized that $\mathcal{C}_{\text{pot}}^{(1)}$ encourages the model to learn more general aspects that are better transferable across chemistry, such as maximizing the distance between non-bonded atoms. The regularization term $\mathcal{C}_{\text{pot}}^{(2)}$ and no regularization, i.e. solely data-driven training, emphasize more specific features found in the particular training set. As such, the generalizability is limited and possible force field inconsistencies can become even more severe.

In summary, the ML-based method DBM is able to generate equilibrated AA molecular configurations based on CG structures. It is a well suited tool to automate backmapping processes as it learns the AA reconstruction from training data and therefore requires little human intervention. Moreover, avoiding unnecessary equilibrations upon reverse-mapping will help to establish a tighter and more consistent link between models at different scales.

The autoregressive reconstruction splits the backmapping task into a sequence of less complex tasks and thereby enables a local environment representation. The locality of DBM is a key feature to achieve remarkable transferability properties, i.e. transferability across thermodynamic state points and across chemical space. As such, DBM offers the perspective to recycle learned local correlations. For example, samples of small molecules can serve as training data for a model that is ultimately deployed on more complex systems. This enables the backmapping of complex molecular structures without necessarily simulating the specific fine-grained system in the first place.

Chapter 6

Backmapping as a Quality Measure for Coarse-grained Models

Central to the bottom-up coarse-graining (CG) approach is the potential of mean force (PMF, Sec. 2.3.2). The PMF is an effective CG potential derived from a reference all-atom (AA) potential, which reproduces the AA probability distribution at the CG resolution [233, 82]. It is often approximated using simple, parameterized potentials that are tuned to reproduce certain distributions observed in the reference AA model [75, 76, 77, 85]. For example, harmonic or tabulated pair potentials between bonded atoms can be tuned to recover the correct bond length and angle distributions, or non-bonded pair potentials can be optimized to reproduce pair distribution functions [72]. However, accurately capturing local or pairwise structural properties does not imply that all cross-correlations and higher-order structures, such as protein tertiary structures, are recovered as well [234, 87, 82]. Therefore, structure-based CG methods could benefit from identifying important many-body effects in order to assess and potentially improve the quality of CG models. In particular, the quality of CG models is typically evaluated at the CG resolution. However, the reduced resolution might hinder the detection of important discrepancies between the AA and CG ensembles.

In this chapter, backmapping is applied to assess the quality of structure-based CG models at the AA resolution. To this end, CG models for Tris-Meta-Biphenyl-Triazine (TMBT) are parameterized using direct Boltzmann inversion (DBI) and iterative Boltzmann inversion (IBI) [57, 91, 92]. At first, the accuracy of the CG models is evaluated in terms of targeted structural distributions at the CG resolution. Afterwards, two backmapping schemes are deployed to reintroduce atomistic details, i.e. deepbackmap (DBM) and a backmapping protocol that relies on energy minimization (EM). In particular, two data sets for the backmapping task are constructed: (1) A data set consisting of AA snapshots projected onto the CG resolution and (2) a data set consisting of snapshots obtained by MD simulations of the CG models. Facilitated by the reintroduced degrees of freedom, the quality of backmapped structures is compared between both test sets and thereby significant discrepancies are revealed.

This chapter summarizes insights obtained during the course of a collaboration

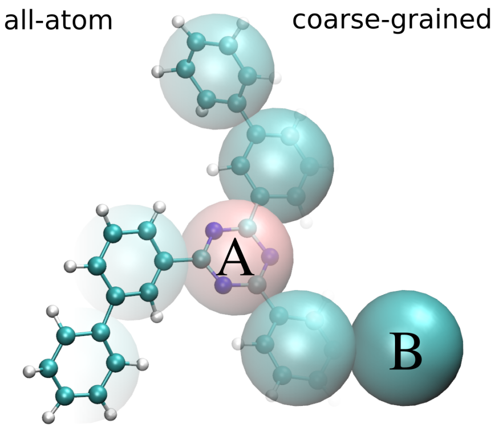

FIGURE 6.1: All-atom (left) and coarse-grained (right) representation of Tris-Meta-Biphenyl-Triazine. The central triazine ring is mapped to one bead of type A and all phenyl rings are mapped to beads of type B.

with Dr. Scherer, Dr. May and Dr. Andrienko. While the underlying project explores organic materials as potential candidates for organic light emitting diodes deploying a multiscale approach, the interesting observations made with respect to the quality assessment of the CG models through backampping is recorded, as they might serve as a starting point for a stand-alone research project in the future. A research article of the presented work is in preparation and will soon be submitted for publication in a peer-reviewed journal.

6.1 Multiscale Modeling of Tris-Meta-Biphenyl-Triazine

This section outlines the structure-based parameterization strategy for the CG models as well as the two deployed backmapping schemes. The proposed method is demonstrated at the example of Tris-Meta-Biphenyl-Triazine (TMBT), a host material for organic light emitting diodes [235]. TMBT is a star-shaped molecule consisting of a central triazine ring and three biphenyl side chains.

6.1.1 Mapping

The CG mapping for TMBT is illustrated in Fig. 6.1. In particular, two bead types are used for the CG representation: The central triazine ring is mapped to one bead of type A and all phenyl rings are mapped to beads of type B. The mapping **M** projects an atomistic configuration **r** to the CG resolution, such that each bead I is positioned at the center of mass $\mathbf{R}_I$ of all atoms i associated with it,

$$\mathbf{R}_I = \mathbf{M}_I(\mathbf{r}) = \sum_{i \in \Psi_I} c_{iI} \mathbf{r}_i, \quad c_{iI} = \frac{m_i}{\sum_{i \in \Psi_I} m_i}, \tag{6.1}$$

where Ψ_I is the set of atomic indices corresponding to bead I, $\mathbf{r}_i$ is the position and m_i the mass of atom i, respectively.

6.1.2 All-atom simulation

AA Simulations are performed using the GROMACS 2019.3 package [226]. The underlying force field is described in [235]. Equilibration runs are carried out for 60 ns in the NPT ensemble at $p = 1.0\,\text{bar}$ using a Parrinello-Rahman barostat with a time step of 1 fs. Production runs are carried out in the NVT ensemble using a velocity rescaling thermostat at 450 K. In particular, production runs are performed for 20 ns at the mean density of preceding NPT equilibration runs. Electrostatic interactions are treated with a smooth particle mesh Ewald method with fourth-order cubic interpolation, 0.12 nm Fourier spacing and an Ewald accuracy parameter of 10^{-5}. A short-range cutoff of $r_{\text{cut}} = 1.3\,\text{nm}$ is used and long-range dispersion corrections for energy and pressure are applied. The simulation box contains 3000 molecules.

6.1.3 Coarse-grained Force Field

The CG force field for TMBT is parameterized based on the AA NVT simulation data. In particular, bonded interactions are parameterized deploying DBI [57], while non-bonded interactions are obtained using IBI [91, 92]. More information on the parameterization schemes can be found in Sec. 2.3.2.

The bonded interaction potentials derived with DBI include two bonds (A-B, B-B), two angles (A-B-B, B-A-B), one proper (B-A-B-B) and one improper (A-B-B-B) dihedral. The latter stabilizes the plane of the central triazine ring and the biphenyl side chains. Distribution functions for all bonded interactions are obtained from AA reference data mapped onto the CG resolution. The obtained interaction potentials are smoothed and tabulated. Proper dihedral interactions are expressed as analytical functions of the Ryckaert-Belleman type, $\sum_{i=0}^{5} c_i \cos\left(180^\circ - \phi\right)$, where c_i are the coefficients of the power expansion. The improper dihedral interactions are modeled by quadratic functions. The coefficients are determined by a least squares fit to the tabulated potentials.

Non-bonded pair interactions between the beads of type A and B are parameterized using 200 steps of IBI in order to match the pair correlation functions $g(r)$ of the CG reference data. All CG potentials are short-ranged with a cutoff of $r_{\text{cut}} = 1.3\,\text{nm}$. In each iteration step, a 200 ps CG NVT simulation at the density of the AA simulation is conducted. A time step of 1 fs is used, and a velocity rescaling thermostat at 450 K is deployed. A simple pressure correction scheme is applied every second iteration by adding a small linear perturbation to the pair potential,

$$\Delta U_{\text{PC}} = -A\left(1 - \frac{r}{r_{\text{cut}}}\right), \quad r_{\text{cut}} = 1.3\,\text{nm}, \tag{6.2}$$

where $A = -\mathrm{sgn}\,(\Delta p)\, 0.1 k_{\mathrm{B}} T \min\,(1, f\Delta p)$, and $\Delta p = p_{\mathrm{i}} - p_{\mathrm{target}}$. A scaling factor $f = 0.001$ is chosen.

After the non-bonded pair potentials are obtained, bonded interactions are rescaled until the distributions of the CG simulation match those of the mapped atomistic simulation. Finally, a 20 ns production run of the CG simulation is performed. The same simulation parameters are deployed as for the IBI iteration steps.

6.1.4 Backmapping

Backmapping of CG TMBT is performed using the machine learning methodology deepbackmap (DBM), which is introduced in Sec. 4 and thoroughly tested in Sec. 5. In addition, a second method that relies on energy-minimization (EM) is applied.

DBM

DBM is trained for 40 epochs with a batchsize of 64 using the same specifications as described in Sec. 5.1.4. The data set consists of four pairs of AA and corresponding CG snapshots, where each snapshot contains 3000 molecules. The energy minimizing regularization term $\mathcal{C}_1$ is used based on the force field of the AA MD simulation.

EM

The EM-based backmapping scheme uses the software package *Versatile Object-oriented Toolkit* (VOTCA) [236]. The backmapping protocol inserts atomistic fragments into the CG structure, such that the centers of mass of atoms are aligned with the corresponding CG bead positions. This initial AA structure is then relaxed by four cycles of EM. The first three cycles are restraint optimizations, i.e. a strong force is introduced to enforce a pinning of the atom positions to their respective CG sites. In the first EM step, only bond-stretching and bending is applied. The second step introduces bond rotations, and in the third/fourth step all interactions are switched on.

6.2 Results

Three different CG models are examined that differ in their bonded interactions: *Model A* includes two angles (A-B-B, B-A-B), one proper (B-A-B-B) and one improper (A-B-B-B) dihedral, while bond lengths are constraint to the average bond length obtained for the reference data, *model B* includes two bonds (A-B, B-B), two angles (A-B-B, B-A-B), one proper (B-A-B-B) and one improper (A-B-B-B) dihedral, and *model C* only includes two bonds (A-B, B-B) and two angles (A-B-B, B-A-B). All models include the same non-bonded pair interactions between the beads of type A and B. The quality of all CG models is first evaluated in terms of structural distributions at the CG resolution. Afterwards, test sets are constructed for the backmapping task:

(1) The in-distribution test set denotes a collection of AA snapshots projected onto the CG resolution. (2) In addition, data sets are constructed consisting of snapshots from MD simulations of the different CG models, which will be referred to as generalization test sets in the following. Both backmapping methods are deployed to all test sets.

6.2.1 Evaluation at the Coarse-grained Resolution

Structural distributions associated with the parameterized interaction potentials can be found in Fig. 6.2. All CG models are able to reproduce the targeted structural distributions of the reference system with remarkable accuracy. However, model A yields a sharply peaked distribution for the bond lengths due to the applied constraints, as shown in panels (a) and (b). Moreover, model C does not recover the distribution functions for the proper and improper dihedrals in panels (e) and (f), which is expected since the corresponding interaction potentials are neglected for this model. In addition, small deviations from the reference system are observed for model C in terms of the pair correlation function $g(r)$ displayed in panels (g) and (h), as well as for the angle (B-A-B) displayed in panel (d). As such, model A and B clearly outperform model C in terms of structural accuracy.

6.2.2 Evaluation at the All-atom Resolution

Backmapping is performed for configurations from two different sources: (1) An in-distribution test set, which denotes AA MD simulation data that is projected onto the CG resolution. This data is used to obtain the baseline accuracy of the backmapping method. (2) A generalization test set, which denotes snapshots obtained by MD simulations based on the CG force fields. To assess and compare the quality of backmapped snapshots for the in-distribution and the generalization test sets, atomistic pair correlation functions and force distributions are analyzed.

Pair Correlation Functions

Selected pair correlation functions obtained with both backmapping schemes are displayed in Fig. 6.3. For readability, only the AA reference system, in-distribution test set and the generalization test set for model A are shown. Similar results for the other CG models can be found in the appendix A.3. Applying DBM to the in-distribution test set yields pair correlation functions that are in excellent agreement with the atomistic reference systems, as can be seen in panels (a), (c) and (e). On the other hand, the EM-based scheme displayed in panels (b), (d) and (f) over-stabilizes the system and therefore yields pair correlations that are more peaked compared to the reference system.

Turning to the results obtained for the backmapped generalization test set reveals that DBM can not maintain its performance observed for the in-distribution test set. The most significant differences are large tails towards small distances in the pair

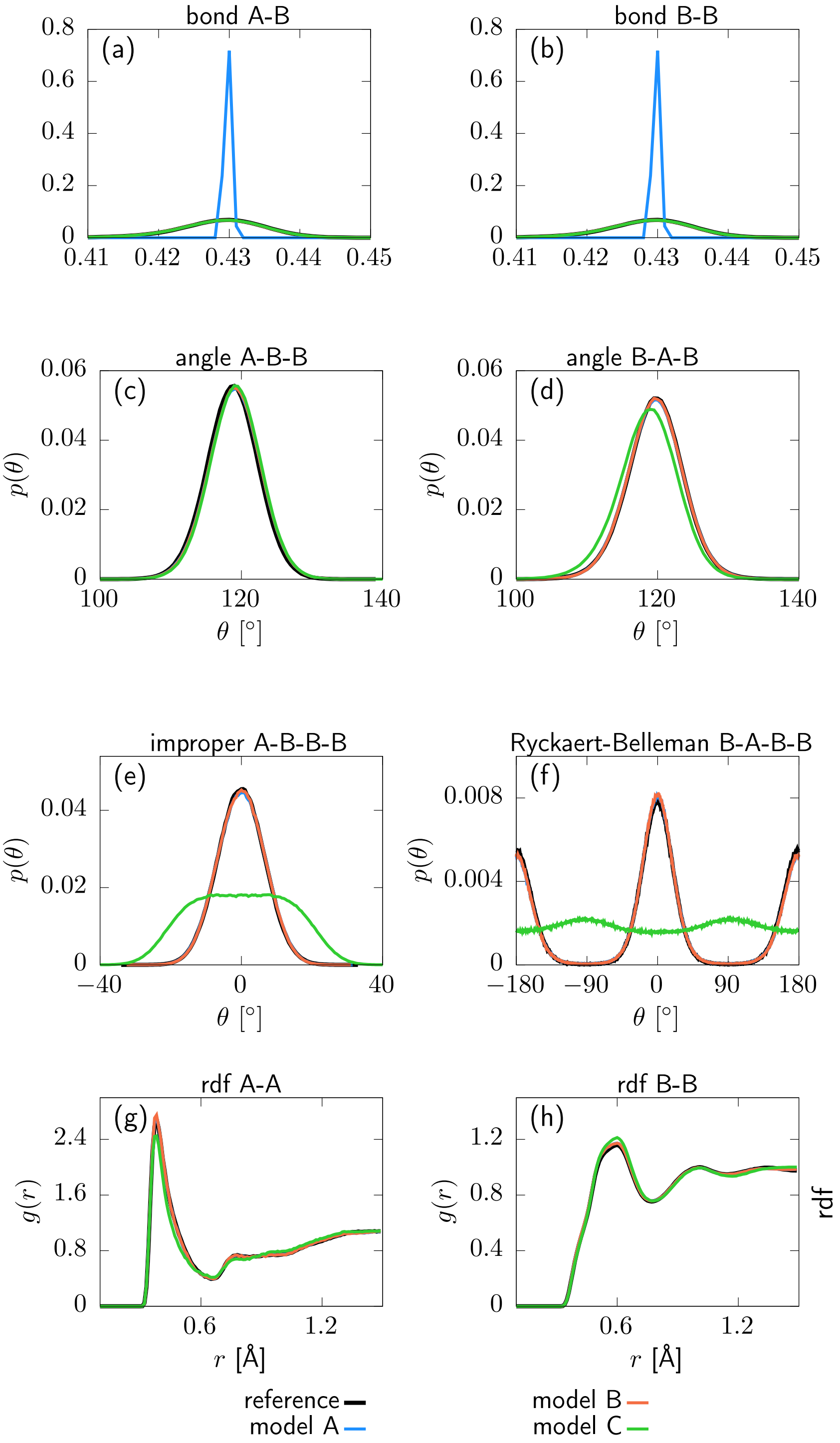

FIGURE 6.2: Structural distribution functions for various force field terms obtained for three different CG models: Model A includes bonded interaction potentials that include two angles (A-B-B, B-A-B), one proper (B-A-B-B) and one improper (A-B-B-B) dihedral, while bond lengths are constraint. Model B includes two bonds (A-B, B-B), two angles (A-B-B, B-A-B), one proper (B-A-B-B) and one improper (A-B-B-B) dihedral. Model C includes two bonds (A-B, B-B), two angles (A-B-B, B-A-B). All models include non-bonded pair interactions between the beads of type A and B. (a) A-B bond, (b) B-B bond, (c) A-B-B angle, (d) B-A-B angle, (e) A-B-B-B improper dihedral, (f) B-A-B-B proper dihedral, (g) radial distribution function $g(r)$ of type A beads, (h) radial distribution function $g(r)$ of type B beads.

	DBM [nm]	EM [nm]
in-distribution	0.0056	0.0423
model A	0.0064	0.0868
model B	0.0063	0.0866
model C	0.0064	0.0884

TABLE 6.1: Root mean-square deviations for in-distribution and generalization test sets computed between backmapped and original CG configurations.

correlation functions indicating steric clashes. On the contrary, the EM scheme yields similar results for the generalization test set compared to the in-distribution test set.

An explanation for the observed results can be found in Fig. 6.4, which displays a superposition of a CG structure and its corresponding backmapped configuration deploying both backmapping schemes. The underlying CG conformation consists of two TMBT molecules that are in close contact to each other. While structural properties of both molecules, such as distances between non-bonded beads, are consistent with the distributions used for parameterization of the CG force field, the specific CG conformation does not allow for an AA reconstruction that (1) is consistent with the CG structure, i.e. atomistic details are reinserted along the CG variables, and (2) has high statistical weight, i.e. a structure with low potential energy. Since DBM is trained with an emphasis on the first requirement, it is not able to fulfill the second requirement, i.e. some inter-atomic distances are too small. On the other hand, the fragment-based scheme violates the first requirement in order to fulfill the second, i.e. the energy minimization shifts the atomistic structure away from the underlying CG configuration in order to avoid close atomic contacts. To underpin these insights, the backmapped structures are projected onto the CG resolution to compute their root mean-square deviation (RMSD) to the original CG configuration. The RMSDs obtained for both backmapping schemes and all three CG models are displayed in Table 6.1. The EM-based backmapping scheme yields RMSDs that are one order of magnitude larger compared to the results obtained with DBM. In summary, MD simulations of all CG models yield CG structures with significant probability that contain cross-correlations inconsistent with the AA ensemble.

Forces

While atomistic pair correlation functions already reveal a discrepancy between the AA and CG ensembles, the AA force field can be used as a quality measure that is more sensitive to steric effects. To this end, the force field used for the AA MD simulation is deployed to calculate forces acting on the atoms. However, as stated in Sec. 2.4, the coarse-to-fine mapping is not unique and a single CG structure corresponds to an ensemble of AA microstates. As such, a direct comparison of forces acting on reference and backmapped particles is not insightful. Therefore, atomistic forces are coarse-grained to enable a more stringent comparison. In particular, the

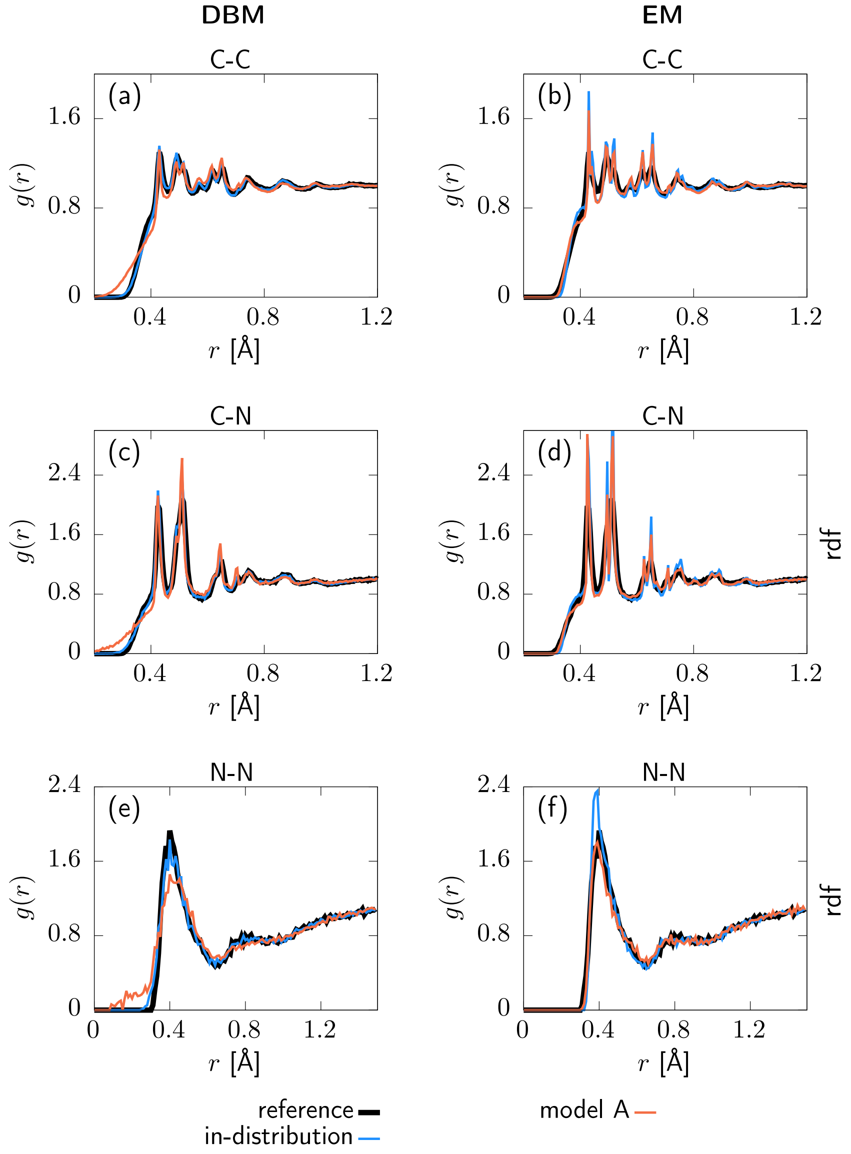

FIGURE 6.3: Pair correlation functions $g(r)$ for the AA reference system, backmapped in-distribution test set and backmapped test set for CG model 1. Results obtained with DBM (left) and EM scheme (right) are displayed, including non-bonded (a)-(b) C-C, (c)-(d) C-N and (e)-(f) N-N correlations.

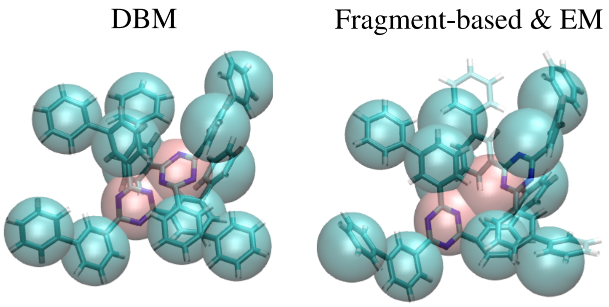

FIGURE 6.4: Superposition of a CG conformation from the generalization test set and backmapped conformation obtained with DBM (left) and EM scheme (right). The CG structure yields too close atomic contacts upon backmapping with DBM, while the AA conformation obtained with the EM scheme is shifted from the CG origin.

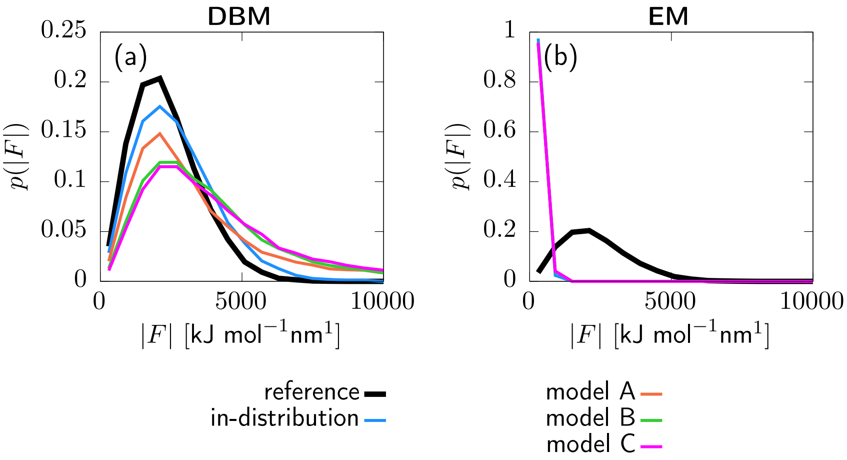

FIGURE 6.5: Force distributions for reference, backmapped in-distribution and backmapped generalization test sets. Backmapping with DBM (left) and the EM scheme (right). Forces are obtained deploying the AA force field and are projected onto the CG resolution.

CG force $\mathbf{F}_I^{\mathrm{AA}}$ is the net force acting on all atoms i associated with bead I,

$$\mathbf{F}_I^{\mathrm{AA}} = \sum_{i \in \Psi_I} \mathbf{f}_i^{\mathrm{AA}}, \tag{6.3}$$

where Ψ_I is the set of atomic indices corresponding to bead I and $\mathbf{f}_i^{\mathrm{AA}}$ is the atomic force acting on atom i.

Fig. 6.5 displays the CG force distributions obtained for the reference, backmapped in-distribution and backmapped generalization test sets. As shown in panel (a), DBM is able to recover the reference forces with high accuracy for the in-distribution test set, which can be regarded as the baseline accuracy of the backmapping method. However, the generalization test sets yield force distributions that differ significantly from the reference. In particular, long tails towards large forces are observed for all CG models indicating steric clashes, i.e. some

	DBM	EM
in-distribution	0.0473	4.9364
model A	0.4571	4.8580
model B	0.5988	4.7915
model C	0.7161	4.8574

TABLE 6.2: Jensen-Shannon divergences for in-distribution and generalization test sets computed between backmapped and reference force distribution. Forces are obtained deploying the AA force field and are projected onto the CG resolution.

atoms are in too close contact with each other. For a more quantitative comparison, Table 6.2 lists the Jensen-Shannon (JS) divergences between the reference and backmapped force distributions. All CG models yield JS divergences that are at least one order of magnitude larger compared to the in-distribution test set. Moreover, a clear ranking for the deployed CG models can be obtained: The best match with the reference force distribution is observed for model A, while the largest discrepancy can be found for model C. This is reasonable, since model C does not take dihedrals into account. On the other hand, force distributions obtained for the EM-based backmapping scheme displayed in panel (b) are not insightful. All distributions are shifted towards significantly smaller forces due to the relaxation and a clear distinction between the models is not possible.

Towards Improving Ensemble Consistency

Evaluating forces based on the AA force field opens new routes towards improving the CG force field parameterization schemes. An evident starting point is the multiscale coarse-graining approach, which is described in Sec. 2.3.2 [93, 60, 94]. The force-matching functional χ aims at matching two kind of CG forces: (1) A projection of AA forces $\mathbf{F}^{\mathrm{AA}}(\mathbf{r})$ onto the CG resolution, which are derived using the reference AA force field for a AA configuration $\mathbf{r}$ and (2) CG forces $\mathbf{F}^{\mathrm{CG}}(\mathbf{M}(\mathbf{r}))$ derived using the parameterized CG force field for a projection $\mathbf{M}(\mathbf{r})$ of the same AA configuration $\mathbf{r}$. Note that the functional χ is therefore evaluated in the AA ensemble,

$$\chi^2[\mathbf{F}^{\mathrm{CG}}] = \frac{1}{3N}\Big\langle \sum_{I=1}^{N} |\mathbf{F}_I^{\mathrm{CG}}(\mathbf{M}(\mathbf{r})) - \mathbf{F}_I^{\mathrm{AA}}(\mathbf{r})|^2 \Big\rangle_{\mathrm{AA}}. \tag{6.4}$$

As such, the actual CG ensemble is not taken into account during parameterization of the CG force field. In order to improve the consistency between the AA and CG ensembles, backmapping could be used to evaluate the CG ensemble in terms of the AA force field. In particular, the functional χ could be augmented

$$\chi^2_{\mathrm{BM}}[\mathbf{F}] = \chi^2 + \frac{1}{3N}\Big\langle \sum_{I=1}^{N} |\mathbf{F}_I^{\mathrm{CG}}(\mathbf{R}) - \mathbf{F}_I^{\mathrm{AA}}(\mathbf{BM}(\mathbf{R}))|^2 \Big\rangle_{\mathrm{CG}} , \tag{6.5}$$

where $\mathbf{BM}(\mathbf{R})$ denotes the backmapping of configuration $\mathbf{R}$ from the CG ensemble. As such, the CG force field would be tuned towards suppressing CG configurations that yield large atomistic forces upon backmapping. Note that computing χ^2_{BM} requires a backmapping scheme $\mathbf{BM}(\mathbf{R})$ that yields consistent reconstructions, i.e. it has to fulfill $\mathbf{M}\big(\mathbf{BM}(\mathbf{R})\big) = \mathbf{R}$.

6.3 Discussion

In this chapter, backmapping is deployed to assess the quality of structure-based CG models at the AA resolution. To this end, CG force fields for TMBT are parameterized using DBI for bonded interactions and IBI for non-bonded interactions. Three different models are parameterized differing in their bonded interactions. It is demonstrated that the CG models reproduce structural properties targeted in the parameterization with remarkable accuracy. Afterwards, test sets are constructed for the backmapping task: (1) An in-distribution test set denotes snapshots obtained in a AA MD simulation that are projected onto the CG resolution. (2) Generalization test sets are constructed consisting of snapshots obtained in MD simulations deploying the CG force fields. While the former is used to assess the baseline accuracy of the backmapping method, a comparison between backmapped in-distribution and generalization test sets yields insights into the quality of the deployed CG models.

Backmapping of CG structures is performed following two different strategies: (1) The machine learning approach DBM and (2) a baseline method that relies on EM are applied. While DBM is able to reproduce AA pair correlation functions for the in-distribution test set with remarkable accuracy, application to the generalization test sets yields AA structures that contain steric clashes, i.e. some atoms are in too close contact with each other. On the other hand, the baseline backmapping method is more robust and maintains its performance for both test sets. However, the baseline method yields pair correlation functions that are overly peaked compared to the atomistic reference due to the relaxation. These findings can be rationalized with respect to two requirement a backmapping scheme has to fulfill: (1) Reconstructed AA details have to be consistent with the underlying CG structure and (2) the backmapped structure has to agree with the Boltzmann distribution. A visual inspection reveals that the generalization test sets contain CG conformations that prohibit reconstructing AA details that fulfill both requirements simultaneously. In particular, DBM generates AA structures that are consistent with the CG structure but consequently display unavoidable steric clashes. The baseline method generates structures with high statistical weight, i.e. low potential energies, but violates the consistency criteria. More specifically, an analysis of the root mean-square deviations between backmapped structures projected to the CG resolution and the original CG configurations reveal a significant shift upon application of the baseline method, while DBM generates AA structures that are close to the given CG configuration.

A more quantitative measure to identify steric clashes is given by the Jenson-Shannon divergence computed between force distributions. In particular, forces acting on the atoms are computed deploying the AA force field and then projected onto the CG resolution. DBM yields a force distribution for the backmapped in-distribution test set that matches the AA reference distribution remarkably well, while distributions for the generalization test sets display long tails towards large forces. Moreover, the JS divergences provide a clear ranking for the quality of the different CG models contained in the generalization test set. Force distributions obtained with the baseline backmapping method are not insightful, since the involved energy minimization yields indistinguishable force distributions that are shifted towards small forces.

Future research might focus on new parameterization strategies for CG force fields that incorporate quality measures at the atomistic resolution. Here, an approach is outlined based on the multiscale force-matching strategy that deploys backmapping to evaluate the CG ensemble in terms of the AA force field. Typically, the force-matching functional is evaluated in the AA ensemble, i.e. it only contains structural information regarding cross-correlations observed in the AA model. However, the CG model is in general not guaranteed to reproduce cross-correlations sufficiently. As such, a force evaluation in terms of the CG ensemble can reveal inconsistencies of the cross-correlations and has therefore the potential to improve the force-matching strategy. In particular, the proposed parameterization scheme aims at suppressing CG configurations that yield large atomistic forces upon backmapping.

Chapter 7

Morphing of Local Statistics: Mapping Through a Resolution Bottleneck

Top-down coarse-grained (CG) models are designed to study the implications of general rules, which are typically inferred from universal physical principles or constructed to reproduce specific phenomena. Unlike bottom-up CG models, top-down models are not build upon a higher resolution model. However, top-down models can still be related to a specific chemistry. To this end, the deployed interaction potentials are tuned in order to reproduce certain properties of a target system, such as density [103], interfacial tension [104] or partitioning of compounds between aqueous and hydrophobic environments [105].

An example of such chemically-specific top-down models is the Kremer-Grest (KG) polymer model with an additional bending potential [112, 113, 114]. A relation to real polymers can be established by matching the experimentally observed Kuhn number, which is a key parameter to characterize a specific polymer chemistry [115, 116]. Such Kuhn scale matched model polymers can be regarded as a special case of structure-based coarse-graining: Controlling the Kuhn number with the parameter for the chain stiffness allows for a reproduction of emergent universal large length- and timescale behavior. However, while this remarkably simple model is able to retain the behavior above the Kuhn scale, particular properties below the Kuhn scale, i.e. local properties, are not expected to resemble the target system [115]. Specifically, solely structure-based CG models on a similar level of resolution are presumed to yield a locally more faithful representation.

In this chapter, a machine learning (ML) method to adjust local properties of molecular structures is introduced. In particular, two distributions of molecular configurations are considered, 1) a distribution of configurations sampled from a top-down CG model, which will be referred to as top-down distribution in the following, and 2) a target distributions, which denotes more faithful representations of the same molecular system. It is assumed that molecular configurations from both distributions share large-scale properties, but differ locally. To improve the quality of the top-down distribution, a ML model is trained to transform it, such that it resembles the target distribution more closely. To this end, the ML model is trained to reproduce local correlations learned from the target distribution, while large-scale properties are maintained. This adjustment of local features based on a target distribution will be referred to as *morphing* in the following.

The motivation for this project is to introduce a two-step backmapping scheme for top-down CG models. A mismatch of local properties on the CG scale between the top-down distribution and a particular target system can impact the quality of backmapped structures, i.e. unphysical artifacts at the higher resolution are expected to occur more frequently. In order to reduce such artifacts already on the CG scale, local statistics of the CG structure are corrected before serving it as an input for the backmapping algorithm.

In the following, the method is applied to two systems: (1) A polymer melt of syndiotactic polystyrene (sPS) sampled with the KG model with tuned bending potential and (2) a condensed-phase system of the alkane tetracosane (TCS) sampled with the Martini force field [117]. The content presented in this chapter is not published yet.

7.1 Method

The method applied in this chapter aims at morphing local features of molecular structures by passing them through a resolution bottleneck. The idea is inspired by the concept of cross-modal learning (CML) known in the ML community [237, 238, 239, 240]. As illustrated in Fig. 7.1 a), CML is used to link sources of information from different domains, for example to perform text-to-image translation. As an instructive example, consider a distribution of strings $\mathcal{A}$ and a distribution of images $\mathcal{B}$. In order to map a string to the domain of images, two autoencoders A and B are trained to encode and decode elements $\mathbf{a}$ and $\mathbf{b}$ sampled from $\mathcal{A}$ and $\mathcal{B}$, respectively. At its core, a link between both distributions can be established by matching the encoded distributions in the latent space. In particular, mapping between $\mathcal{A}$ and $\mathcal{B}$ is performed by cross-connecting the encoder e and decoder d of both models, i.e. $d_B(e_A(\mathbf{a}))$ for text-to-image translation. However, connecting the latent distributions of both models, i.e. the information-bottlenecks, is challenging and subject to current research [241].

As illustrated in Fig. 7.1 b), a similar approach to CML is applied to link two distributions of molecular structures $\mathcal{X}$ and $\mathcal{Y}$. It is assumed that molecular configurations of both distributions display similar large-scale features, but differ in their local properties. In the following, $\mathcal{X}$ denotes the distribution of a top-down CG model, while $\mathcal{Y}$ denotes a target distribution representing more faithful molecular structures, for example obtained by a structure-based CG method. In order to map from $\mathcal{X}$ to $\mathcal{Y}$, a ML-based function g is introduced to learn the transformation $g(\mathcal{X}) \approx \mathcal{Y}$. To this end, both distributions are linked at a lower-resolution, i.e. at an

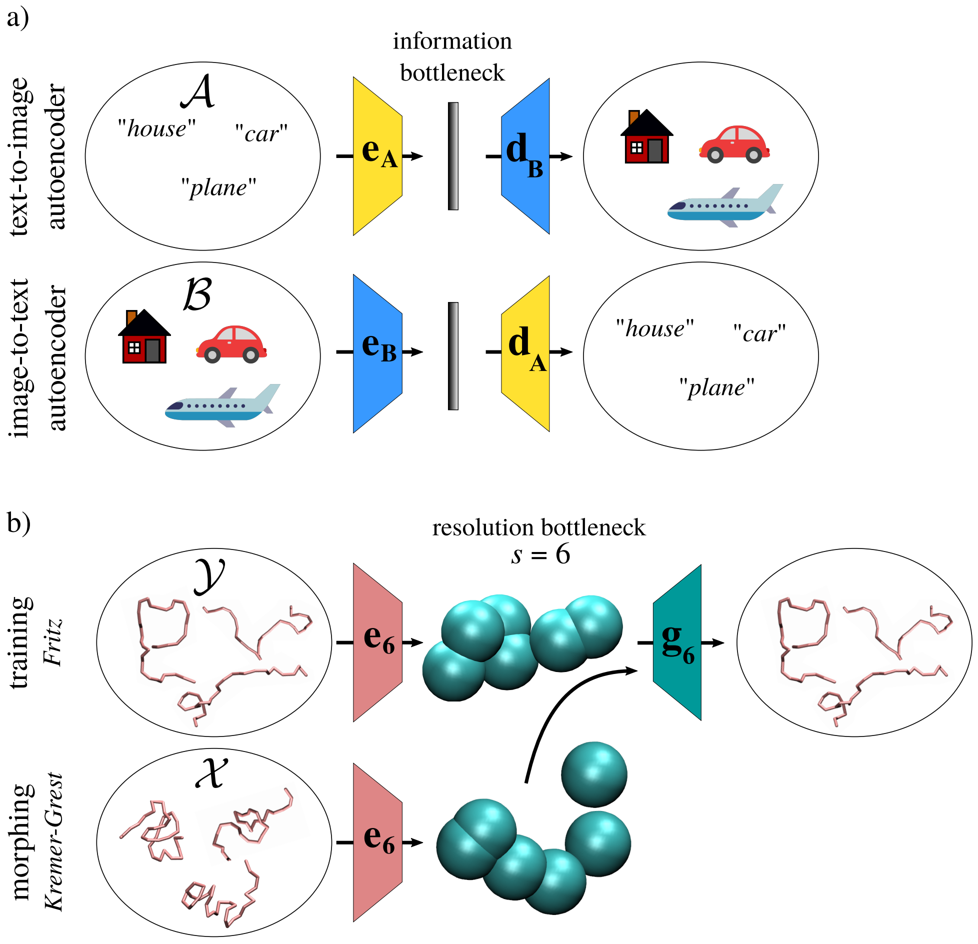

FIGURE 7.1: a) Illustration of cross-modal learning: An autoencoder A is trained to encode and decode samples from a text-distribution $\mathcal{A}$, and another autoencoder B is trained to encode and decode samples from an image-distribution $\mathcal{B}$. In order map from one domain to the other, for example to achieve text-to-image translation, the encoder e and decoder d of both models can be cross-connected, i.e. both distributions are matched at the information-bottleneck.
b) Illustration of the morphing approach: Molecular structures from a distribution $\mathcal{X}$ are mapped through a resolution bottleneck in order to reinsert local features learned from a target distribution $\mathcal{Y}$. To this end, an encoder e_s is applied, which reduces the degrees of freedom by a factor s, and a backmapping model g_s is trained to reinsert details. Importantly, training of g_s is fixed to the target distribution $\mathcal{Y}$. Afterwards, the trained model is transferred to $\mathcal{X}$. Choosing the value for s is a tradeoff between the complexity of the backmapping task and the impact of the morphing.

information-bottleneck. In particular, a simple encoder e_s is chosen that reduces the number of particles n by a factor s

$$e_s : \mathbb{R}^{3n} \rightarrow \mathbb{R}^{3n/s}, \tag{7.1}$$

, i.e. e_s denotes a fine-to-coarse mapping of the coordinates. Specifically, e_s computes the center of mass for groups of s particles. Afterwards, encoded structures are backmapped to the original resolution deploying the ML model g_s, such that $g_s(e_s(\mathcal{Y})) = \mathcal{Y}$. To this end, g_s is trained with the deepbackmap (DBM) approach, which is introduced in Chpt. 4. Importantly, training of g_s is fixed to the target distribution $\mathcal{Y}$. Afterwards, the trained model is transferred to $\mathcal{X}$ to perform the morphing $g_s(e_s(\mathcal{X})) \approx \mathcal{Y}$, i.e. to reinsert local correlations into the CG structures $e_s(\mathcal{X})$ learned from $\mathcal{Y}$.

The value for the coarse-graining factor s scales the extent to which local features are varied. Assuming that the backmapping scheme yields a perfect reconstruction, the mapping $g_s(e_s(\mathcal{X}))$ is expected to yield a more accurate reproduction of $\mathcal{Y}$ the larger s becomes, since the reinsertion of details becomes less restricted by the CG representation $e_s(\mathcal{X})$. However, larger values of s lead to a more complex backmapping exercise. As such, choosing the value for s is a tradeoff between the complexity of the backmapping task and the impact of the morphing.

Two different morphing schemes A and B, respectively, are tested in this work. Scheme A refers to the basic backmapping protocol outlined in Chpt. 4, i.e. the method deploys forward sampling to obtain an initial high-resolution structure, which is further refined applying Gibbs sampling. In contrast, scheme B skips the forward sampling step and utilizes the original high-resolution structure drawn from $\mathcal{X}$ as an initial structure, i.e. only Gibbs sampling is applied.

The proposed method is data driven, as the morphing is learned from training data, and does not require to parameterize a force field for the given target distribution. However, the quality of morphed structures can be improved by incorporating a simple potential energy during training of the ML model that penalizes certain configurations in terms of bond lengths, angles and non-bonded distances. An illustration of such a potential energy landscape can be found in Fig. 7.2. In particular, a harmonic potential of the form

$$U(\phi) = \begin{cases} a(\phi - \phi_{\text{min}})^2, & \phi < \phi_{\text{min}} \\ a(\phi - \phi_{\text{max}})^2, & \phi > \phi_{\text{max}} \\ 0, & \text{otherwise}, \end{cases} \tag{7.2}$$

is applied as bonded interaction, where ϕ represents bond lengths or angles, respectively, a is a scaling factor, and $\phi_{\text{min}}/\phi_{\text{max}}$ are threshold values of the potential.

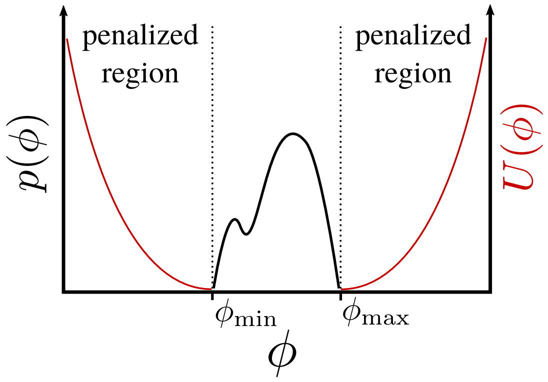

FIGURE 7.2: Illustration of a simple energy landscape for bonded interactions to improve the quality of morphed molecular structures. ϕ represents bond lengths or angles, ϕ_{min} and ϕ_{max} are the minimum and maximum values obtained from the distribution $p(\phi)$, and $U(\phi)$ is a harmonic potential to penalize regions below ϕ_{min} or above ϕ_{max}.

Similarly, a harmonic potential for non-bonded distances d

$$U(d) = \begin{cases} a(1 - \frac{d}{d_{\mathrm{min}}})^2, & d < d_{\mathrm{min}} \\ 0, & \text{otherwise}, \end{cases} \tag{7.3}$$

is introduced, where d_{min} is the minimum distance for non-bonded particles. The values for the minimum and maximum distances/angles are obtained from the target distribution.

7.2 Set-up and Reference Data

The morphing approach is applied to a sPS polymer melt sampled with the KG model with tuned bending potential [112, 113], and a condensed-phase system of the alkane tetracosane sampled with the Martini force field [117]. In addition, a higher resolution model is deployed for each system to obtain locally more faithful molecular structures. Specifically, the molecular sPS model by Fritz *et al.* is used to obtain a target distribution for the KG model [227], and an all-atom (AA) simulation of TCS with the GROMOS-96 force field is performed for the Martini model [242]. For a direct comparison of the top-down and target distributions, the higher resolution configurations are projected onto the resolution of the top-down models.

7.2.1 Kremer-Grest Model: Syndiotactic Polystyrene

The KG model is a standard model for computer simulations of polymeric systems [112, 113]. It is designed to study generic polymer properties with an emphasize on computational efficiency and simplicity. As outlined in Sec. 2.3.2, the KG model is a bead-spring model, where consecutive beads are connected via strong nonlinear springs, i.e. the FENE potential (Eq. 2.42), and mutual interactions between all beads are modeled via a truncated Lennard-Jones potential (Eq. 2.43). The deployed interaction potentials are tuned such that topological constraints found in real polymeric systems are reproduced, i.e. chain backbones are prohibited to pass through each other. As such, the KG model is able to yield large scale entanglement properties that are characteristic for long-chain polymers. In order to modify the stiffness of the polymer chains, an additional bending potential can be introduced (Eq. 2.44), which is scaled by a prefactor κ [114].

Matching at the Kuhn Scale

An important characteristic of many polymeric systems is their universal large-scale behavior that manifests in scaling relations. Specifically, the *mean square end-to-end distance* $\langle R_e^2 \rangle$ of a polymer chain scales with the number of beads N, i.e. $\langle R_e^2 \rangle \propto N^{2\nu}$. In a melt state, polymers adopt the characteristics of a random-walk and $\nu = \frac{1}{2}$. However, local interactions of real polymers introduce correlations between monomers that ultimately increase $\langle R_e^2 \rangle$. In order to account for such correlations, the results known for ideal chains, i.e. the random-walk behavior $\langle R_e^2 \rangle = l^2 N$, requires a correction

$$\langle R_e^2 \rangle = C_\infty l^2 N, \tag{7.4}$$

where l is the bond length between consecutive monomers and C_∞ is *Flory's characteristic ratio*. Note that C_∞ depends on the local stiffness of the polymer chain, i.e. emergent large-scale properties are influenced by microscopic details.

The crossover from local, chemistry specific to universal, random-walk behavior is characterized by the *Kuhn length b*. It is defined by mapping the real chain onto an equivalent ideal chain with n segments of length b that reproduces $\langle R_e^2 \rangle$ and the contour length $L = Nl$, i.e.,

$$\langle R_e^2 \rangle = b^2 n, \tag{7.5}$$

$$L = nb. \tag{7.6}$$

A key parameter to characterize a specific polymer chemistry is the *Kuhn number* n_k. It is a dimensionless parameter, which defines the number of Kuhn segments within a cube of length b,

$$n_k = \rho_k b^3, \tag{7.7}$$

where ρ_k is the number density of Kuhn segments. It is observed that n_k systematically correlates with emergent properties, such as the entanglement length [115, 116]. As such, n_k can be used to link experimentally observed polymers with model polymers.

While the Kuhn number n_k is material specific and depends on atomic details, it is not straightforward to infer its dependence on the deployed interaction potentials. However, Everaers *et al.* have found a direct relation between the chain stiffness κ of the KG model and the implied Kuhn number n_k,

$$b(\kappa) = b^{(0)} + \Delta b, \tag{7.8}$$

$$\Delta b = 0.77\sigma(\tanh(-.03\kappa^2 - 0.41\kappa + 0.16) + 1), \tag{7.9}$$

where $b^{(0)}$ is the bare Kuhn length in the absence of excluded volume interactions and σ is the bead diameter [116]. Given the Kuhn length b, the corresponding Kuhn number n_k can be inferred from Eq. 7.7.

In summary, the KG model with additional bending potential offers a one parameter model that covers a wide range of experimentally relevant polymers. However, while this remarkably simple model is able to retain the behavior above the Kuhn scale, no particular effort is put into reproducing the correct local properties. Therefore, local properties are likely to differ from the target system [115].

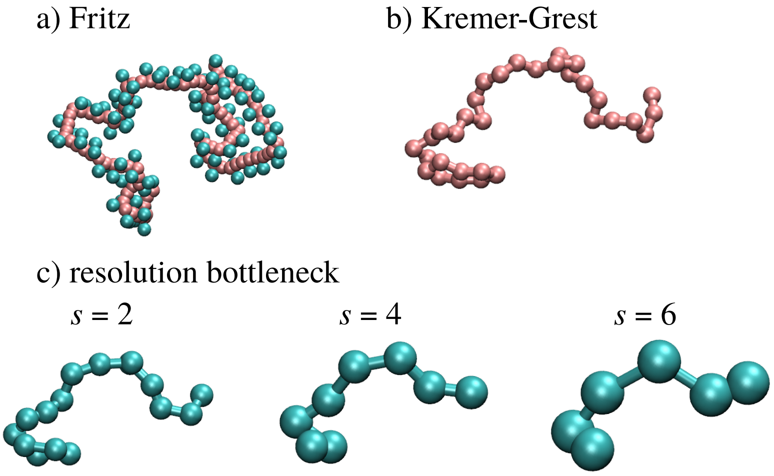

FIGURE 7.3: Illustration of the different resolutions for sPS. a) Fritz model, b) KG model, and c) resolution bottleneck for the morphing approach.

Sampling

To underpin the above statement, the CG model for sPS by Fritz *et al.*, which is already discussed in Sec. 5.1.1, is deployed to obtain a target distribution. The model, which will be referred to as *Fritz* model in the following, is parameterized based on detailed AA simulations of stereoregular PS sequences in vacuum and reproduces the target thermodynamic properties with remarkable accuracy [227]. A simulation of the Fritz force field is carried out in the NPT ensemble at $T = 496$ K using the molecular dynamics package GROMACS 5.0 [226]. Temperature and pressure of the system are controlled using the velocity rescaling thermostat and the Parrinello-Rahman barostat. An integration time step of 1 fs is used and samples are recorded every 2.5 ns. The simulation box contains 24 molecules consisting of 96 monomers each. As illustrated in Fig. 7.3, the Fritz model has a higher resolution compared to the KG model. In order to analyze the discrepancies between the Fritz model and KG model, the former is mapped onto the resolution of the KG model. Following [115], three polystyrene monomers are mapped onto a single KG bead, which is positioned according to the center of mass of the corresponding monomers.

Snapshots of the KG model with an equivalent number of polymers and chain size are sampled by an NVT simulation performed with *ESPResSo++* [243]. The standard parameters for the KG model are deployed, i.e. the bead density $\rho = 0.85\sigma^{-3}$, the distance at which the FENE potential diverges $R = 1.5\sigma$ and the bond length $l = 0.965\sigma$. Note that the model is athermal since all interaction potentials scale with $k_B T$. While [115] only provides values for the stiffness parameter κ associated with isotactic and atactic polystyrene, the mean value of both, i.e. $\kappa = 0.8815$, is deployed in this study as an educated guess to model syndiotactic polystyrene. In general, the stiffness parameters listed in [115] are only valid at the reference temperature $T = 413$ K. Here, a higher temperature was chosen to sample the Fritz model, since sPS undergoes a phase transition from a melt to a crystalline phase at $T \approx 450$ K [224]. However, the authors of [115] state that static melt properties are relatively insensitive to changes of the temperature. The bead diameter is set to $\sigma = 1.0$ throughout the simulation, but distances are rescaled afterwards in order to match the particle density of the Fritz model.

7.2.2 Martini Model: Tetracosane

An additional test of the morphing procedure is demonstrated for the Martini model. As described in Sec. 2.3.2, the Martini force field is a generic CG force field for a wide range of soft matter systems with an emphasis on biomolecules [117, 118, 119, 120]. The parameterization of the force field is based on the top-down approach for non-bonded interactions and on the bottom-up approach for bonded interactions. The Martini model is widely used due to its robust transferability across soft matter systems. However, the price for the transferability of the Martini model is a less accurate reproduction of structural features for particular systems [122].

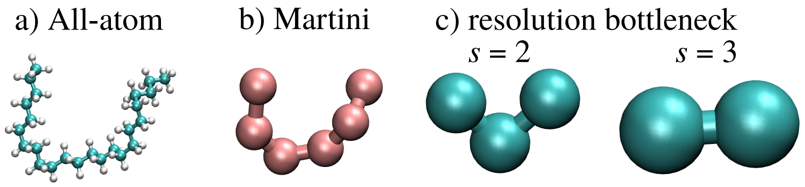

FIGURE 7.4: Illustration of the different resolutions for TCS. a) AA model, b) Martini model, and c) resolution bottleneck for the morphing approach.

Sampling

The Martini model is used for a MD simulation of TCS, which is an alkane hydrocarbon with the structural formula $H(CH_2)_{24}H$. An illustration of TCS can be found in Fig. 7.4. Applying the Martini mapping rules, the Martini representation for a TCS molecule consists of six beads of the apolar type C. A MD simulation of a TCS liquid based on the Martini force field is carried out in the NPT ensemble at $T = 400$ K using the molecular dynamics package GROMACS 5.0 [226]. Temperature and pressure of the system are controlled using the velocity rescaling thermostat and the Parrinello-Rahman barostat. An integration time step of 10 fs is used and samples are recorded every 0.5 ns. The simulation box contains 168 molecules.

To obtain a target distribution, an AA simulation of TCS is performed. To this end, the GROMOS-96 force field is used and topologies are generated by AUTOMATED TOPOLOGY BUILDER [242, 228]. The MD settings are equivalent to the settings of the Martini simulation, except for a reduced integration time step of 1 fs. The AA simulation is projected onto the resolution of the Martini model by mapping every four carbon atoms and associated hydrogens to their center of mass.

7.3 Results

In the following, the impact of the morphing is evaluated. To this end, the morphing model is trained to reproduce local features of the target distribution. Afterwards, the trained model is applied to morph configurations obtained with the top-down model. For the evaluation, structural distributions and free energy landscapes are analyzed. For each system, the top-down, target and morphed distributions are compared. For both test systems, morphing is performed using both schemes outlined in 7.1, i.e. scheme A (forward sampling and Gibbs sampling) and scheme B (only Gibbs sampling).

Morphing of the local statistics for snapshots obtained with the KG model is performed for $s = 2, 4, 6$, as illustrated in Fig. 7.3 c). The training set for DBM consists of 22 snapshots obtained with the Fritz model projected onto the KG resolution. 500 samples are used for the evaluation. For the Martini model, morphing is performed for $s = 2, 3$, as shown in Fig. 7.4 c). The training set for DBM consists of 16 snapshots obtained with the GROMOS simulation projected onto the Martini resolution. For the evaluation, 104 samples are used.

7.3.1 Large-scale Characteristics

Table 7.1 summarizes large-scale properties for the sPS system, such as the mean square end-to-end distance $\langle R_e^2 \rangle$, radius of gyration $\langle R_g^2 \rangle$ and contour length L. As expected, the Kuhn scale matched KG model yields similar large-scale characteristics as the Fritz model. The biggest impact upon morphing is observed for $s = 6$ in

	$\sqrt{\langle R_e^2 \rangle}$ [nm]	$\sqrt{\langle R_g^2 \rangle}$ [nm]	L [nm]
Fritz model	6.06 ± 0.03	2.51 ± 0.02	22.92 ± 0.2
KG model	6.01 ± 0.03	2.43 ± 0.02	22.58 ± 0.2
scheme A, $s = 2$	6.02 ± 0.03	2.43 ± 0.02	22.63 ± 0.2
scheme A, $s = 4$	6.05 ± 0.03	2.43 ± 0.02	22.36 ± 0.2
scheme A, $s = 6$	6.01 ± 0.03	2.39 ± 0.02	20.87 ± 0.2
scheme B, $s = 2$	6.01 ± 0.03	2.43 ± 0.02	22.56 ± 0.2
scheme B, $s = 4$	6.00 ± 0.03	2.44 ± 0.02	22.53 ± 0.2
scheme B, $s = 6$	6.01 ± 0.03	2.45 ± 0.02	21.81 ± 0.2

TABLE 7.1: Large-scale characteristics of the Fritz model, KG model and morphed structures deploying both morphing schemes A and B with different morphing scales $s = 2, 4, 6$.

terms of the contour length L. However, the overall impact of DBM on the large-scale properties is not significant.

7.3.2 Structural Distributions

The discrepancy between the Fritz and KG model becomes more apparent for local structural features, as illustrated in Fig. 7.5. Panel (a) and (b) display distribution functions for the bond length. While the KG model yields polymers with a sharp bond length distribution that has a peak at 0.72 nm, polymers obtained with the Fritz model display a broader distribution peaked at 0.79 nm. The impact of the morphing on the bond length distribution varies and depends on the morphing scale s as well as the applied morphing scheme. The best results deploying morphing scheme A are obtained for $s = 2$, where the bond length distribution of the morphed structures match remarkably well with the reference Fritz model. Larger values of s deteriorate the morphing capability in terms of the bond lengths. Regarding scheme B, the best results are obtained for the intermediate morphing scale $s = 4$, while $s = 2$ has an negligible impact on the distribution and $s = 6$ yields a distribution that is too broad.

The angle distributions shown in panels (c) and (d) reveal that the KG as well as the Fritz model yield polymeric structures that cover a wide range of angles between consecutive bonds. However, the angle distribution for the Fritz model displays a single peak at $\approx 135^\circ$, while the distribution for the KG model has two peaks: A major peak at $\approx 130^\circ$ and a side peak at $\approx 70^\circ$. As such, the KG model puts higher statistical weight on small angles $< 100^\circ$ compared to the Fritz model. Both morphing schemes are able to suppress small angles for $s = 6$. Smaller values of s reduce the impact of the morphing on the angle distribution. Especially, morphing scheme B at $s = 2$ has no noticeable impact.

Panels (e) and (f) display the pair correlation $g(r)$ between non-bonded beads. Polymer melts generated with the KG model yield a sharply peaked pair correlation function, while the $g(r)$ for melts obtained with the Fritz model are less peaked and reveal a shorter minimum distance between non-bonded beads. Both morphing schemes smooth the pair correlation of KG structures. The best match with the reference Fritz model is obtained for scheme A and $s = 2$.

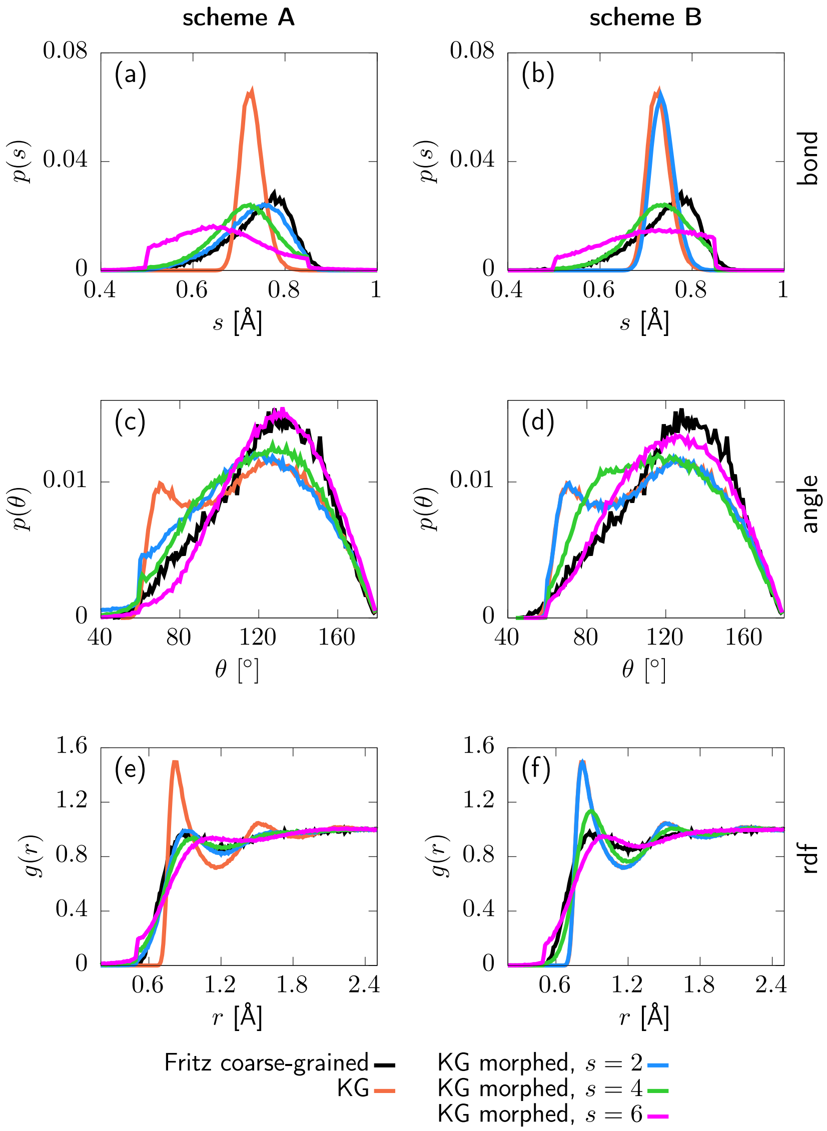

FIGURE 7.5: Canonical distributions for sPS at the KG resolution. (a) and (b) bond length distribution, (c) and (d) angle distribution, (e) and (f) radial distribution function $g(r)$. Left: morphing scheme A. Right: Morphing scheme B

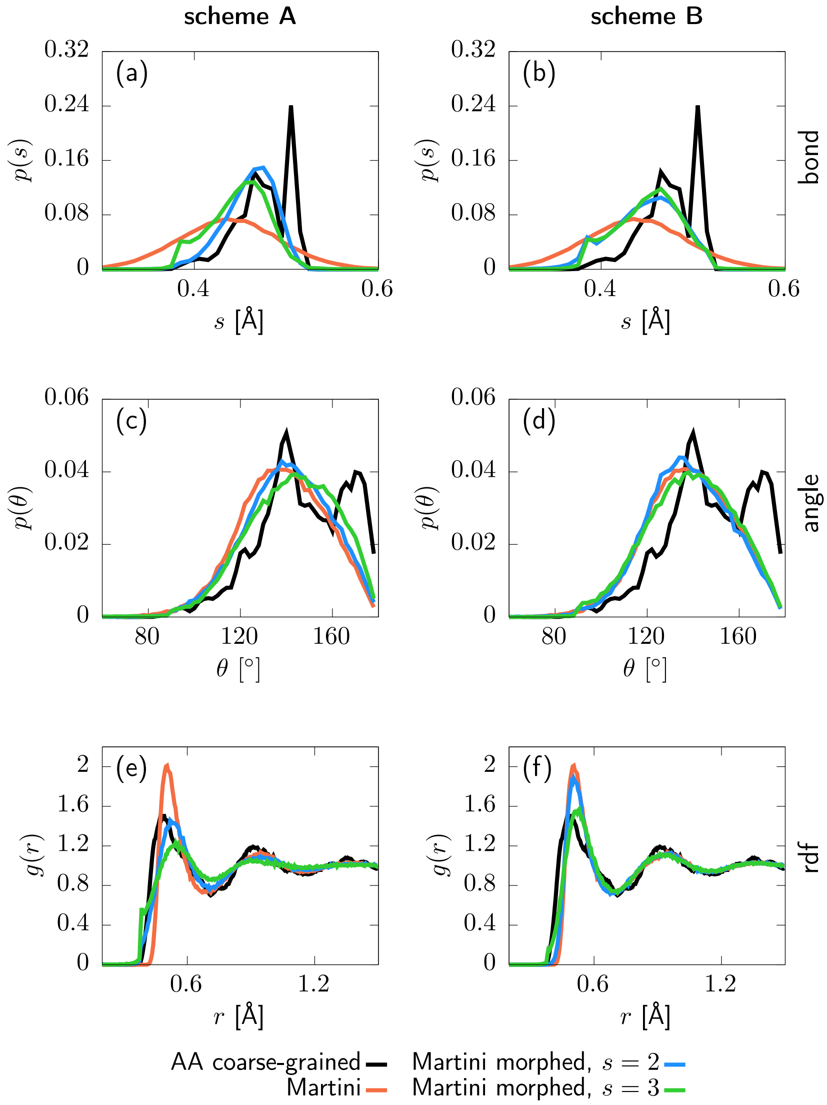

FIGURE 7.6: Canonical distributions for TCS at the Martini resolution. (a) and (b) bond length distribution, (c) and (d) angle distribution, (e) and (f) radial distribution function $g(r)$. Left: morphing scheme A. Right: Morphing scheme B

Fig. 7.6 displays structural distributions for TCS. The distributions of the bond length shown in panels (a) and (b) differ significantly for the target and Martini structures. While the Martini model yields a broad and Gaussian shaped bond length distribution that extends from $\approx$ 0.30 nm to $\approx$ 0.60 nm, the distribution obtained for the AA simulation projected onto the Martini resolution is more structured and compressed. It extends from $\approx$ 0.38 nm to $\approx$ 0.54 nm and has two peaks at $\approx$ 0.47 nm and $\approx$ 0.52 nm. While DBM is able to compress the bond length distribution to the observed range of the target system, it is not capable of recovering the specific structure of the distribution. Specifically, it fails to reproduce the second peak at $\approx$ 0.52 nm. The best match with the target distribution is visually obtained for morphing scheme A and $s = 2$.

The angle distributions displayed in panels (c) and (d) follow a similar trend. While the distributions obtained for the Martini and the projected GROMOS simulation data cover a similar range, the later is more complex. The Martini distribution displays a single peak at $\approx 135°$ and suppresses large angles $> 170°$, whereas the projected AA configurations yield two peaks at $\approx 140°$ and $\approx 170°$. None of the morphing models is able to correct the angle distribution sufficiently. The best result is obtained visually for scheme A and $s = 3$, which slightly shifts the distribution towards larger angles.

The pair correlation function $g(r)$ depicted in panels (e) and (f) is sharply peaked for the Martini model. The projected AA configurations yield a less peaked distribution and a slightly shorter minimum distance between non-bonded beads. Most morphing schemes are able to smooth the pair correlation function except for scheme B and $s = 2$. The best match is visually obtained for scheme B and $s = 3$.

7.3.3 Sketch-map Free Energy

For a further analysis of the configuration space, free energies surfaces (FESs) are computed. However, the high dimensionality of the molecular configurations prohibits a direct visualization. Therefore, the free energy is computed in terms of low dimensional collective variables that characterize the state of the molecule. Here, the focus is set to local properties rather than large-scale chain statistics. Therefore, dimensionality reduction is applied to generate low-dimensional representations for a set of local features. To this end, sketch-map (SM) coordinates are deployed to construct FESs. In particular, local environments $\mathcal{H}$ centered along the molecular chain are constructed and the pairwise distance between two such environments is encoded using a similarity kernel $k(\mathcal{H}, \mathcal{H}') = \mathbf{p}(\mathcal{H})\mathbf{p}(\mathcal{H}')$ based on the normalized many-body SOAP representation $\mathbf{p}(\mathcal{H})$ [231]. In order to obtain a global similarity kernel $k(a, b)$ for two molecular chains a and b, the covariance matrix $C_{ij}(a, b) = \mathbf{p}(\mathcal{H}_i^a)\mathbf{p}(\mathcal{H}_j^b)$ is mapped to a single scalar value using an average kernel [232].

Fig. 7.7 displays the obtained FESs for the sPS system expressed in SM coordinates. Landmarks for the SM coordinates are obtained for 524 polymer chains sampled from both, the KG as well as the Fritz model. For each panel, 12000 polymer chains are projected onto the SM space guided by the landmarks. The projections are used to construct a two-dimensional histogram with 50 bins along each dimension. The discretized populations N_i for each bin i are then used to compute free energies

$$F_i = -k_{\mathrm{B}}T\ln(N_i) + \text{const.} \tag{7.10}$$

The SM FES obtained for the Fritz model is depicted in panel (a). The diverse set of conformations obtained in the melt yields a single blob in the SM space. However, a trend within the projected blob is observed when the positions of selected chain structures are analyzed, as shown for a few examples within Fig. 7.7: The majority of sPS chains corresponding to a mean square end-to-end distance close to the observed average are mapped to the blob center (example (2)). The second SM axis a_2 correlates with the extension of the chain, i.e. elongated chains are mapped to small values of a_2 (example (3)) while collapsed chains are mapped to large values of a_2 (example (1)). The role of first SM axis a_1 is more ambiguous. While large values of a_1 can be associated with u-shaped chains (example (4)), small values of a_1 can not be associated with specific shapes of the sPS chains.

The FES for the KG model displayed in panel (b) clearly differs from the FES obtained for the Fritz model. The overall shape of the obtained blob is more compressed and has an elliptic shape. The intersection between the two FESs is small, as the majority of KG chains are mapped to significantly smaller values for the SM axis a_1 compared with the Fritz model.

The morphed structures yield SM FESs that vary in shape. While both morphing schemes yield distinct results, a systematic shift of the obtained blobs towards larger values for the second SM axis a_2 is observed for larger values of the morphing factor s for both schemes. The best match with the Fritz model is visually obtained for scheme A and $s = 2$. On the other hand, deploying morphing scheme B with $s = 2$ has no significant impact on the FES, i.e. the results for the original KG model are reproduced. This observation agrees well with the analysis of structural distributions. Example structures (5)-(8) illustrate the impact of the morphing deploying different morphing models. While example (5) displays an original KG structure, examples (6)-(8) illustrate corresponding morphed structures. Deploying scheme A and $s = 2$ (example (6)) has visually only a minor impact on the overall chain structure. However, the resulting modifications are sufficient to reposition it to an area associated with a high statistical weight in terms of the FES obtained for the Fritz model. Applying scheme A and $s = 6$ (example (7)) yields an overly smoothed chain structure that is mapped to an area associated with low statistical weight. Morphing scheme B and $s = 4$ (example (8)) results in a false structure, as indicated by an overstretched bond. Consequently, it gets mapped to an area not covered by the Fritz distribution.

Table 7.2 displays the Jenson-Shannon (JS) divergences computed for the free energy distributions of the morphed structures and the reference Fritz model. The reported JS divergences underpin the results obtained from visually inspecting Figs. 7.5 and 7.7, i.e. the smallest value is obtained for morphing scheme A and $s = 2$.

	scheme A	scheme B
s = 2	0.436	4.654
s = 4	1.953	0.646
s = 6	3.589	3.495

TABLE 7.2: Jenson-Shannon divergences computed for the free energy distributions of morphed structures and reference Fritz configurations. The Jenson-Shannon divergence between distributions of original KG configurations and reference Fritz configurations is 4.661.

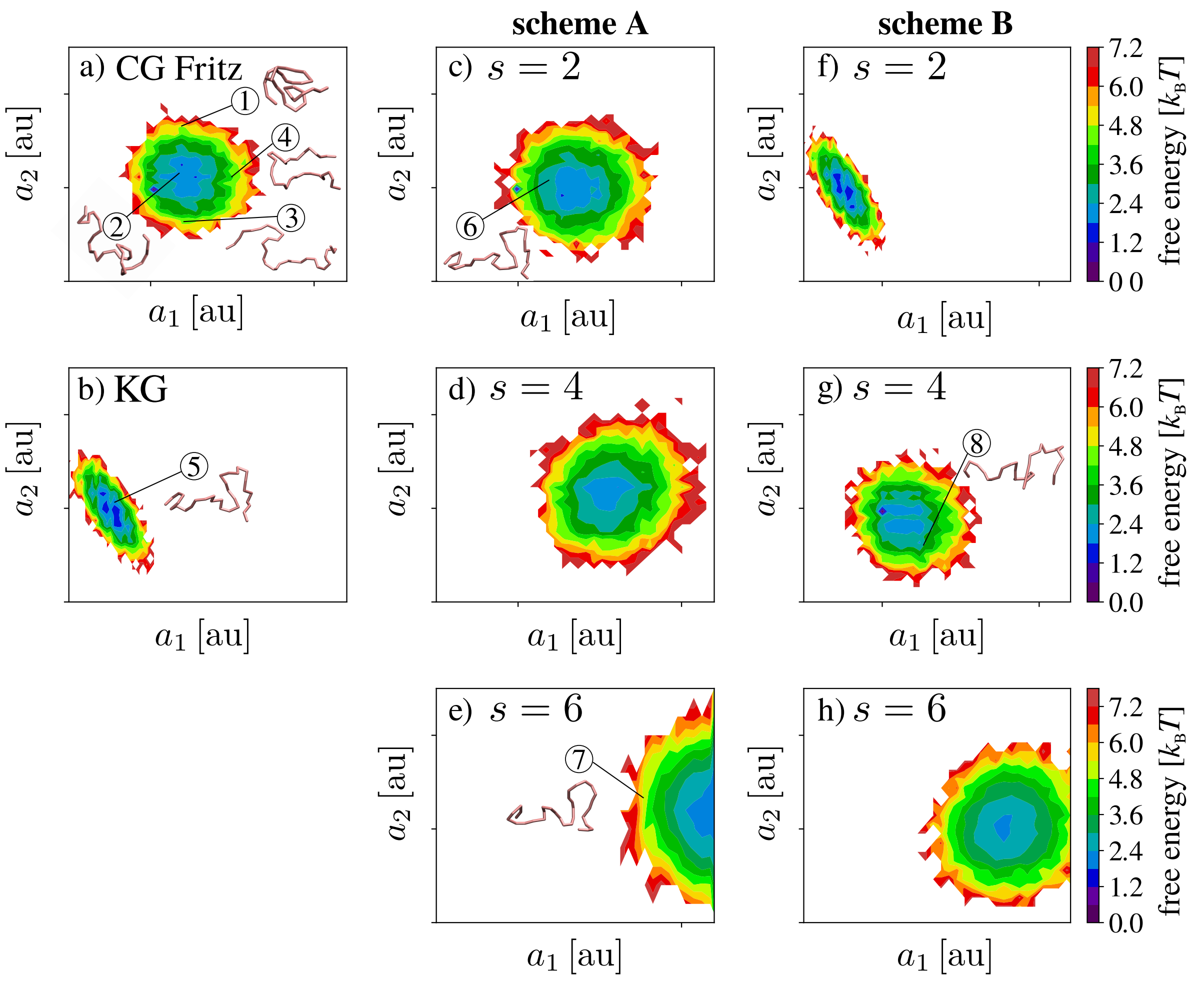

FIGURE 7.7: Free energy landscapes in sketch-map coordinates for (a) the projected Fritz model, (b) KG model and (c)-(h) morphed structures deploying both morphing schemes A and B for $s = 2, 4, 6$.

The FESs for TCS are depicted in Fig. 7.8. Landmarks are obtained for 672 molecules sampled from both, the Martini as well as the AA simulation projected onto the Martini resolution. For each FES 16800 molecules are projected onto the SM space guided by the landmarks. Relative populations are computed over 50 bins along each dimension. The configurations obtained for the projected AA model (panel (a)) yield a densely distributed FES that covers the upper third of the displayed range for the second SM axis a_2. Further analysis of selected molecular conformations reveal that the area associated with the highest occupation at the center of the displayed range of a_1 correspond to u-shaped molecules (example (2)). Smaller values of a_1 can be associated with stretched molecular conformations (example (1)), while larger values of a_1 can be associated with rather rigid conformations containing a kink (example (3)). The occupied region corresponding to the smallest value of a_2 can be associated with collapsed molecular conformations (example (4)).

The FES obtained for the Martini model displayed in panel (b) covers a broader range compared to the target distribution. While most of the probability mass is still centered at larger values for a_2, almost the full range of a_2 is covered. Note that the occupied area of the projected AA distribution is completely covered by the Martini distribution indicating that the Martini model covers a broader area in the configuration space. Example structures (5) and (7) drawn from areas not occupied by the AA model correspond to a condensed conformation and a zig-zag structure, respectively.

Comparing panels (a) and (b) reveals the challenges for the morphing task: The broad area occupied in the configuration space deploying the Martini model has to be compressed and projected onto those regions covered by the AA model and thereby ideally reproducing the correct relative populations. Analyzing panels (c)-(f) indicate that this task is only partly successful. All morphing models are able to shift the probably mass towards larger values of a_2, i.e. pushing it closer to the target distribution. While this procedure is successful for some conformations, such as example (6), which is morphed from (5), it fails for others, such as example (8), which is morphed from (7).

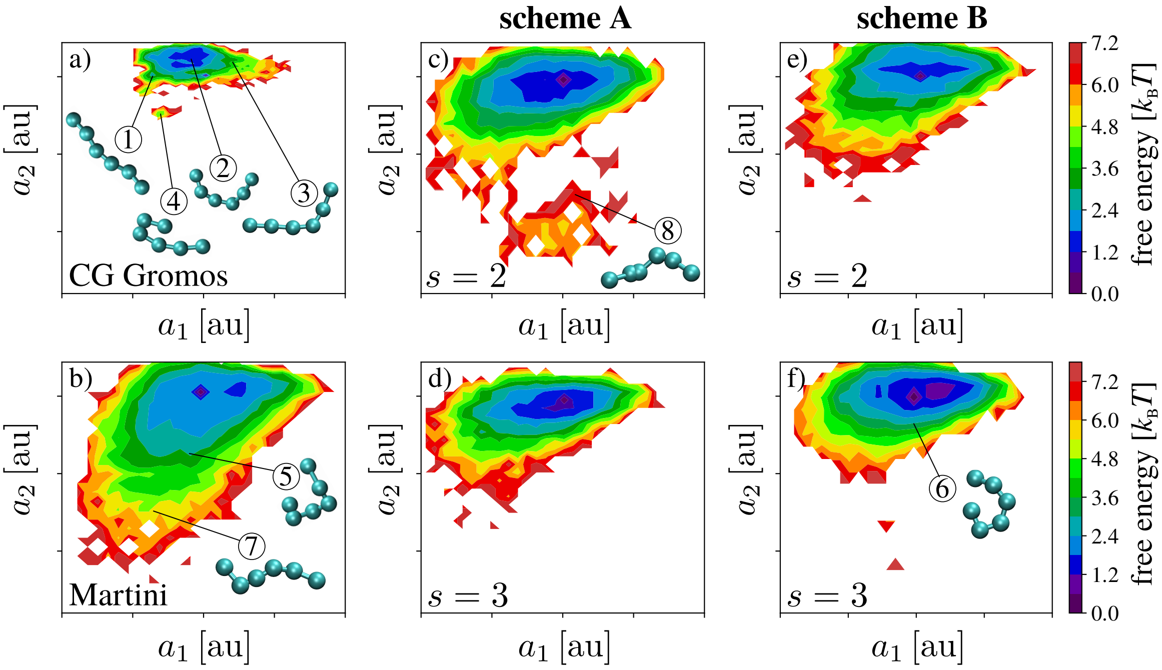

FIGURE 7.8: Free energy landscapes in sketch-map coordinates for (a) the projected AA model, (b) Martini model and (c)-(g) morphed structures deploying both morphing schemes A and B for $s = 2, 3$.

Identifying the best performing morphing model via visual inspection of the structural distributions in Fig. 7.6 and the FESs in Fig. 7.8 is challenging. Therefore, the JS divergence between the free energy distribution for the target and the morphed structures is displayed in Table 7.3. According to the JS divergence, scheme B and $s = 3$ yields the best match.

	scheme A	scheme B
s = 2	2.338	1.213
s = 3	2.925	1.197

TABLE 7.3: Jenson-Shannon divergences computed for the free energy distributions of morphed structures and projected AA configurations. The Jenson-Shannon divergence between distributions of original Martini configurations and projected AA configurations is 2.239.

7.3.4 Backmapping

The motivation to morph local statistics stems from the idea to reduce artifacts upon backmapping due to a mismatch of local properties between the distributions obtained for the top-down CG model and a particular target system. Therefore, the impact of the morphing on the quality of backmapped molecular configurations is investigated.

For the backmapping of sPS melts from the KG resolution to the original resolution of the Fritz model, DBM is trained on 10 snapshots obtained with the Fritz model to reintroduce missing degrees of freedom. Specifically, each CG bead at the KG resolution maps to three Fritz beads of type A and three Fritz beads of type B. DBM is trained using regularization $\mathcal{C}_{\text{pot}}^{(1)}$ based on the force field of Fritz *et al.*. After training, backmapping is performed for CG Fritz configurations, KG configurations, and morphed KG configurations. Morphed structures are obtained with the best performing morphing model, i.e. scheme A and $s = 2$.

Fig. 7.9 displays selected structural distributions for the reference Fritz model and backmapped configurations. The baseline accuracy of DBM is probed by its ability to backmap CG Fritz configurations. DBM is capable of reproducing bond length distribution with remarkable accuracy, as illustrated exemplary for the A-B bond depicted in panel (a). Distributions for the angles of backmapped structures, shown in panels (b) and (c) are slightly too broad and fail to reproduce the correct height of the main peak. Similar issues are observed for the dihedrals of backmapped structures displayed in panel (d). However, the overall accuracy of intramolecular structural distributions is satisfactory. The pair correlation function $g(r)$ depicted in panels (e)-(f) reveals a discrepancy between reference and backmapped structures in terms of distances between non-bonded beads.

The quality of backmapped structures from the KG model does not significantly differ from the baseline quality. In other words, no further artifacts upon backmapping of the KG model can be observed compared to backmapping of the CG Fritz structures. Consequently, the influence of the morphing on the quality of backmapped structures is negligible.

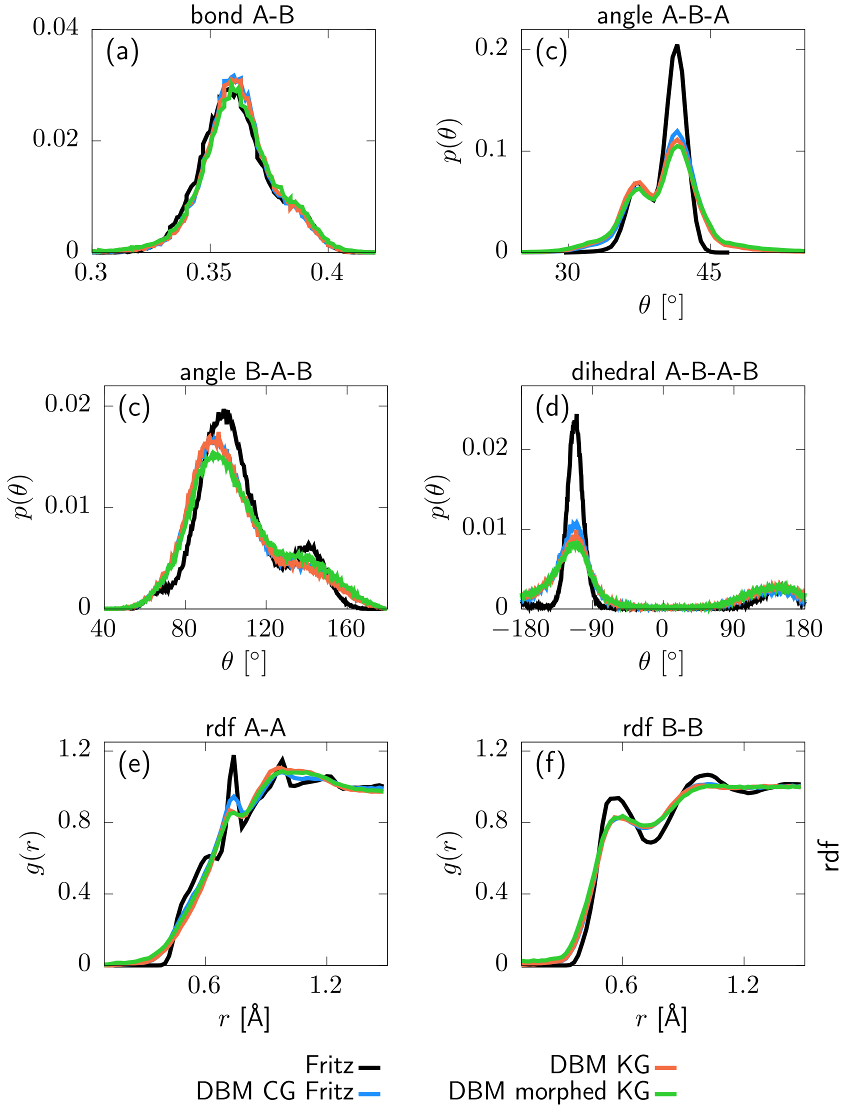

FIGURE 7.9: Canonical distribution for configurations backmapped to the Fritz resolution. (a) bond A-B, (b) angle A-B-A, (c) angle B-A-B, (d) dihedrals A-B-A-B, (e) radial distribution function for non-bonded beads of type A, (f) radial distribution function for non-bonded beads of type B. Backmapping is performed for projected Fritz configurations (blue), KG configurations (red), and morphed KG configurations (green). Morphed structures are obtained with the best performing morphing model, i.e. scheme A and $s = 2$.

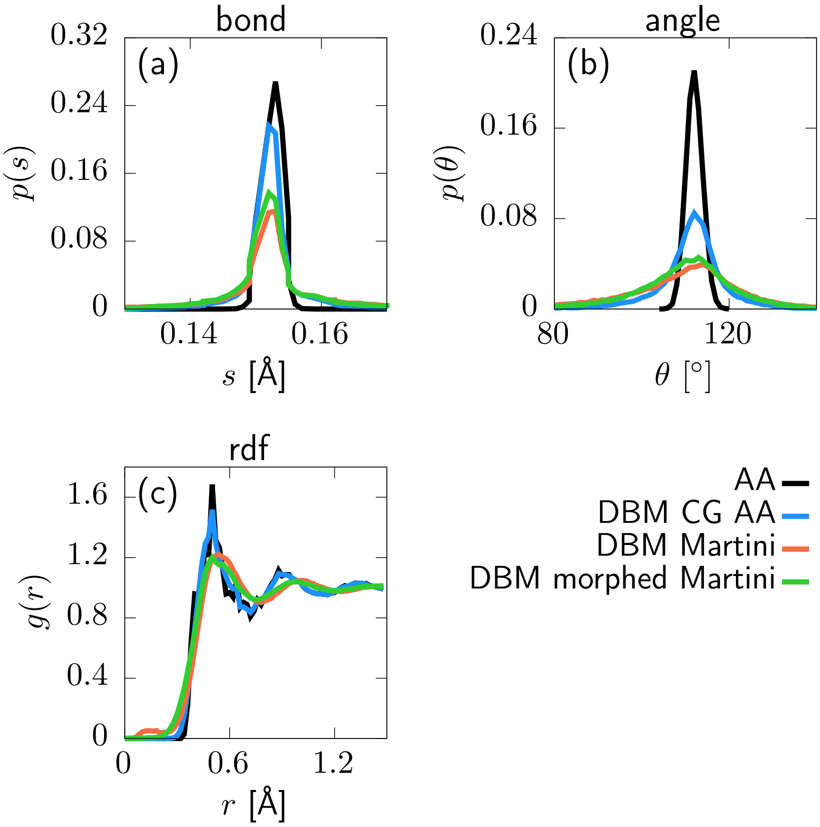

FIGURE 7.10: Canonical distribution of the carbon atoms for configurations backmapped to the AA resolution. (a) bond length distribution, (b) angle distribution and (c) pair correlation function $g(r)$. Backmapping is performed for projected AA configurations (blue), Martini configurations (red), and morphed Martini configurations (green). Morphed structures are obtained with the best performing morphing model, i.e. scheme B and $s = 3$.

In order to backmap the Martini structures, DBM is trained on 16 snapshots of the AA model. Regularization $\mathcal{C}^{(1)}_{\text{pot}}$ based on the GROMOS-96 force field is applied. After training, backmapping is performed for projected AA configurations, Martini configurations, and morphed Martini configurations. Morphed structures are obtained with morphing scheme B and $s = 3$.

Fig. 7.10 displays structural distributions of the carbon atoms for the reference AA and backmapped configurations. The baseline accuracy of DBM is probed by its ability to backmap projected AA configurations. The bond length distribution depicted in panel (a) is reproduced with high accuracy, whereas the angle distribution shown in panel (b) is slightly too broad. However, the small range of angles of the reference AA simulation has to be emphasized. The pair correlation function $g(r)$ displayed in panel (c) is reproduced with high accuracy.

The quality of structures backmapped from the Martini model differs significantly from the baseline quality. Morphing of the Martini model does only yield a minor improvement of the quality: The peaks for the bond length and angle distributions slightly increase upon morphing. Most noticeable, the $g(r)$ indicates that too small distances of non-bonded carbon pairs are suppressed upon morphing.

7.4 Discussion

In this chapter, a ML-based approach to adjust local properties of molecular configurations is introduced. The method aims at improving the quality of structures obtained with chemically-specific top-down models that already capture the correct large-scale behavior of a target system, but differ locally.

In order to correct local discrepancies, molecular configurations are mapped through a resolution bottleneck. In particular, molecular structures are projected onto a lower resolution, and DBM is used to reinsert degrees of freedom. Importantly, DBM is trained solely on structures of a more faithful target distribution and is afterwards transferred to configurations obtained with the top-down model. Therefore, local details learned from the target distribution are inserted into the top-down structures, which is referred to as morphing of local properties.

Two different morphing schemes are probed: Scheme A consists of forward sampling and additional Gibbs sampling, while scheme B starts from the original top-down structure and deploys Gibbs sampling only. It is observed that scheme B has a smaller impact on local properties compared to scheme A. This is reasonable, as scheme A has to generate local features from scratch, while scheme B starts from local features obtained with the top-down model, which might hinder the morphing.

Moreover, the extent to which local features are varied can be controlled by the resolution of the bottleneck. In particular, the number of degrees of freedom in the resolution bottleneck is reduced by a constant factor s. For large values of s, the reinsertion of details becomes less restricted by the representation at the resolution bottleneck, which enables larger variations. However, the larger the value of s the more complex the exercise for the morphing model becomes, as the dependencies between particles, which the morphing model has to learn, also increases with s. Therefore, choosing the value for s is a tradeoff between the complexity and the impact of the morphing. In this study, small values of s yield superior results than large values in most cases.

The morphing approach is tested on Kuhn scale matched KG sPS melts and liquids of the alkane TCS obtained with the Martini model. The sPS melts obtained with the KG model yield similar large-scale characteristics as the higher resolution and solely-structure based model by Fritz *et al.* However, Kuhn scale matching does not take local features below the Kuhn scale into account. As such, local structural distributions of both models differ, which is demonstrated by projecting the melt structures obtained with the Fritz model onto the KG resolution. In particular, the Fritz model yields a broader range of bond lengths compared to the KG model. Furthermore, the Fritz model suppresses small angles and smooths the pair correlation function. Morphing of KG configurations is performed by DBM, which is trained on the Fritz distribution. While morphing has no significant impact on large-scale characteristics, such as the mean square end-to-end distance, it is able to reconstruct local structural features that agree remarkably well with the Fritz distribution. For a higher-order investigation of local properties, FESs are computed in SM coordinates. The analysis of the obtained FESs underpins the aforementioned results.

TCS liquids obtained with the Martini model are compared to AA simulations using the GROMOS-96 force field, which are projected onto the Martini resolution. The GROMOS configurations yield more complex structural distributions compared to the Martini liquids. In particular, bond length and angle distributions obtained for the GROMOS model display multiple peaks and are more compressed than the distributions obtained for the Martini model. Moreover, a analysis of the SM FESs indicate that the occupied region in configuration space is more compact for the GROMOS model compared to the Martini model. To adjust local features, morphing models are trained on the GROMOS distribution and are transferred to the Martini configurations. Unfortunately, none of the morphing models is able to correct local features sufficiently. It can be hypothesized that the discrepancy between the Martini and the GROMOS model are too significant, such that morphing of local properties is not sufficient to match both distributions, i.e. the distributions do not match at the resolution bottleneck.

This project aims at introducing a two-step backmapping scheme for top-down CG models, where local statistics of the CG structure are corrected before it is served as an input for a backmapping algorithm. To assess the impact of the morphing on the quality of backmapped structures, backmapping of morphed KG and Martini structures is performed with DBM. Specifically, DBM is trained to increase the resolution of KG structures to the level of the original Fritz model and Martini structures to the AA level. Only a minor impact of the morphing on the quality of backmapped structures is observed for both systems. In particular, backmapping of KG structures and CG Fritz structures already yield similar distributions of local structural features without morphing. This can be rationalized by the robust transferability of the backmapping model DBM, which was observed in Chpt. 5. It can be hypothesized that strong local interactions, such as covalent bonds, at the higher resolution yield local correlations that separate from larger scales. Therefore, local correlations learned by DBM transfer well across the CG configuration space and backmapping of KG structures yield similar local structural properties compared to backmapping of CG Fritz structures. On the other hand, small differences in the distributions of structural properties are observed between backmapped Martini and CG GROMOS TCS liquids. However, the morphing model for TCS reproduces the complex structural distributions of the GROMOS model only with limited accuracy, such that morphing does not improve the quality of the backmapped liquids significantly.

Chapter 8

Temporal Coherent Backmapping of Molecular Trajectories

MD simulations evolve a molecular system in time and produce a trajectory, i.e. a discretized path in phase space. Typically, consecutive frames of the trajectory are separated by a fixed time step, which dictates the level of temporal resolution. Computing time averages over a trajectory yields structural or thermodynamic properties, such as radial distribution functions or average energies. However, temporal information stored in the trajectory can be used to compute dynamic properties as well. In particular, time correlations can be used to link simulation results to experimental observables. Examples include (1) the diffusion constant, which can be computed as the integral of the velocity auto-correlation [71], (2) (infrared) absorption spectra, which are related to the auto-correlation function of the total dipole moment [244, 245], and (3) scattering functions that can be related to Fourier transforms of the van Hove correlation function, which is a time-dependent pair correlation function [246, 247]. Note that some important dynamic properties, such as the dynamic structure factor, require atomistic details for a comparison with experimental data [248, 249]. While time correlation functions are central to the analysis of dynamic properties, typical reverse-mapping strategies are frame-based, i.e. each molecular snapshot of the trajectory is treated separately [37, 44, 46, 47, 208, 250]. Such backmapping schemes are not temporally aware and correlations between consecutive frames are only maintained via coarse-grained (CG) variables. Consequently, reintroduced degrees of freedom between consecutive frames might decorrelate locally. As such, time correlation functions based on local, atomistic descriptors might not be reliable for such frame-based backmapping strategies.

In this chapter, a new method to perform temporally coherent backmapping of molecular simulation trajectories is introduced. In particular, temporal coherent backmapping refers to reproducing shifts of atomic positions between consecutive frames that agree with the all-atom (AA) reference system. The proposed method aims at both, generating well-equilibrated molecular structures for each individual frame, while maintaining temporal coherence within a series of frames. To this end, a ML model is deployed that reconstructs a molecular structure leveraging information from previous simulation frames. In particular, the model is conditioned on the

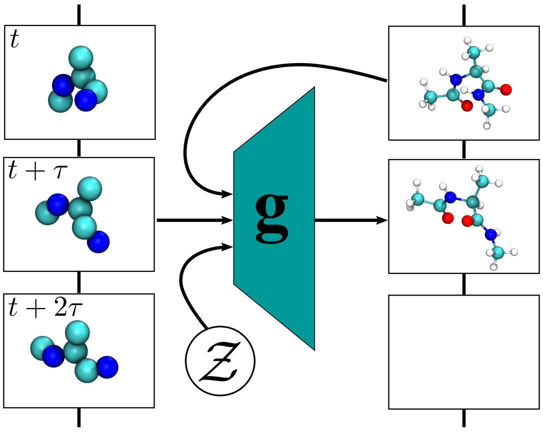

FIGURE 8.1: Illustration of the temporal coherent backmapping approach. Consecutive trajectory frames are spaced by time τ. The ML model generates an atomistic frame $\mathbf{r}_t$ at time t based on the previous atomistic state $\mathbf{r}_{t-\tau}$, the current coarse-grained frame $\mathbf{R}_t$ and latent sample $\mathbf{z}$ from a prior distribution $\mathcal{Z}$.

current coarse- and previous fine-grained state, as illustrated in Fig. 8.1. In contrast to the previously deployed GAN-based method deepbackmap (DBM), a variational autoencoder (VAE) is used for this task.

The method is applied to two biomolecular systems: Alanine dipeptide (ADP) and the mini protein chignolin (CLN). For each system two test sets are constructed: (1) An *in-distribution* test set denotes projections of reference AA trajectories onto the CG resolution. One part of this data set is used to train the backmapping model, while the other part is used to evaluate its baseline accuracy. In particular, the accuracy of the backmapping model is analyzed regarding its ability to reproduce structural and dynamic properties of the reference system. (2) A *generalization* test set is constructed by performing CG simulations deploying approximate force fields generated with CGSchNet, which is a ML-based approach for molecular coarse-graining [251]. The trained backmapping model is transferred to this data set to analyze the CG model on the AA resolution.

The work presented in this chapter stems from a collaboration with Kirill Shmilovich, Moritz Hoffmann and Nick Charron. The project originates from the long program *Machine Learning for Physics and the Physics of Learning* at the Institute for Pure & Applied Mathematics that was held from 09.04.19 to 12.08.19 at the University of California, Los Angeles. This work will soon be submitted for publication in a peer-reviewed journal. A preprint can be found at the free distribution service *arXiv* [252]:

Kirill Shmilovich, Marc Stieffenhofer, Nick Charron, Moritz Hoffmann
Temporally coherent backmapping of molecular trajectories from coarse-grain to atomistic resolution
arXiv preprint:2205.05213
DOI: 10.48550/ARXIV.2205.05213

8.1 Method

In the following, the proposed method is outlined. In addition, Markov state models are introduced as a framework to analyze dynamic properties of molecular systems.

8.1.1 Backmapping Approach

The proposed method is similar in spirit to the previously used method DBM (Sec. 4), but differs in some aspects including the molecular representation, the incorporation of previous states and the architecture of the ML model. While this section emphasizes the distinctions to DBM, a detailed description of the applied ML model can be found in [252].

Molecular Representation

The ML model g generates all atoms of a molecular snapshot in one step, i.e. not autoregressively. Therefore, the local environment representation used for DBM does not apply and molecular representations fed to the model have to capture the structure entirely. In particular, atoms and beads are represented as smooth densities expressed on a discretized grid due to voxelization. Note that the center of mass is removed for each molecule in order to ensure that it is fully enclosed by the grid representation. To avoid overlaps of particle densities that could deteriorate the spatial resolution, each particle is placed in its own feature channel, i.e. a molecule containing N particles with positions $\mathbf{r} \in \mathbb{R}^{3N}$ is represented as a four-dimensional tensor $\mathcal{E}(\mathbf{r}) \in \mathbb{R}^{N \times s \times s \times s}$, where s is the grid size. Note that g also generates voxelized molecular representations $\hat{\mathcal{E}}$. However, these voxel representations can be transformed into Cartesian coordinates $\hat{\mathbf{r}} = m(\hat{\mathcal{E}})$, where m denotes a sequence of differentiable operations. Further information on the voxel and coordinate representation can be found in Sec. 4.3.

Incorporating the Previous State

Central to the proposed method is the incorporation of the previous state in order to achieve temporal coherence between trajectory frames. To this end, the input for the ML model g at time t is augmented with the previous AA state at time $t - \tau$, where τ is the lag time between frames. In particular, the input for the generator g consists of the current CG frame $\mathcal{E}(\mathbf{R}_t)$ and the previous AA frame $\mathcal{E}(\mathbf{r}_{t-\tau})$, where $\mathbf{R} \in \mathbb{R}^{3N}$ and $\mathbf{r} \in \mathbb{R}^{3n}$ denote the coordinates of the N beads and n atoms, respectively. While both, $\mathbf{R}_t$ and $\mathbf{r}_{t-\tau}$, are taken from the reference trajectory during training, AA coordinates $\hat{\mathbf{r}}_{t-\tau} = m\Big(g\big(\mathcal{E}(\mathbf{R}_{t-\tau}), \mathcal{E}(\hat{\mathbf{r}}_{t-2\tau})\big)\Big)$ generated by g in the previous step are used during prediction. As such, trajectories are backmapped autoregressively. The seed for this autoregressive procedure, i.e. the initial AA frame at $t = 0$, is selected from a presampled library of AA configurations based on the root-mean-square deviation at the CG resolution.

Variational Autoencoder

A variational autoencoder (VAE, Sec. 3.4.2) architecture is used instead of the generative adversarial approach [190]. To this end, an encoder network $e\big(\mathcal{E}(\mathbf{r}_t)\big)$ is introduced to generate latent samples $\hat{\mathbf{z}}_t \in \mathbb{R}^d$ based on the current target frame $\mathbf{r}_t$, where d is the dimension of the latent space. The decoder $g\big(\mathcal{E}(\mathbf{R}_t), \mathcal{E}(\mathbf{r}_{t-\tau}), \hat{\mathbf{z}}_t\big)$ is then trained to reconstruct the current state $\mathbf{r}_t$ given the low dimensional embedding $\hat{\mathbf{z}}_t$. As such, the model can be trained end-to-end based on a reconstruction loss and does not rely on an additional critic network.

While the encoder e is indispensable during training, it is omitted during inference. Instead, the latent sample $\mathbf{z}$ is drawn from a prior distribution in order to provide a source of randomness. This non-deterministic approach is an important aspect for the backmapping task, since each CG structure is associated with an ensemble of microstates. In particular, $\mathbf{z}$ is drawn from a 10-component Gaussian Mixture Model (GMM) fitted to the latent distribution implied by the encoder instead of the assumed prior $p(\mathbf{z}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The GMM ensures that the decoder operates within densely sampled latent space regions [253].

Cost-function

The cost-function $\mathcal{C}$ used to train the model end-to-end consists of multiple parts. In particular, a term $\mathcal{C}_{\text{recon vox}}$ is introduced to enforce reconstruction of the spatially voxelized particle densities,

$$\mathcal{C}_{\text{recon vox}}(\mathbf{r}_t, \hat{\mathcal{E}}) = \frac{1}{s^3 n} \, ||\mathcal{E}(\mathbf{r}_t) - \hat{\mathcal{E}}||_2^2. \tag{8.1}$$

Similarly, a term $\mathcal{C}_{\text{recon pos}}$ is introduced to encourage the exact reconstruction of positions

$$\mathcal{C}_{\text{recon pos}}(\mathbf{r}_t, \hat{\mathbf{r}}_t) = \frac{1}{3n} \, ||\mathbf{r}_t - \hat{\mathbf{r}}_t||_2^2. \tag{8.2}$$

In addition, the reproduction of inter-particle distances is targeted by a term $\mathcal{C}_{\text{EDM}}$ that computes the mean squared error between the Euclidean Distance Matrices (EDM) of the target configuration $\mathbf{r}$ and reconstructed configuration $\hat{\mathbf{r}}$,

$$\mathcal{C}_{\text{EDM}}(\mathbf{r}_t, \hat{\mathbf{r}}_t) = \frac{1}{2n^2} \, ||EDM(\mathbf{r}_t) - EDM(\hat{\mathbf{r}}_t)||_2^2. \tag{8.3}$$

In order to achieve consistency between the backmapped structure $\hat{\mathbf{r}}$ and the given CG configuration $\mathbf{R}$, the CG mapping M is applied to introduce a reconstruction loss $\mathcal{C}_{\mathrm{CG}}$ at the CG resolution, i.e.

$$\mathcal{C}_{\mathrm{CG}}(\mathbf{R}_t, \hat{\mathbf{r}}_t) = \frac{1}{3N} \, ||\mathbf{R}_t - M(\hat{\mathbf{r}}_t)||_2^2. \tag{8.4}$$

Moreover, the AA force field is deployed in $\mathcal{C}_{\mathrm{energy}}$ to calculate the mean squared error between the potential energies for the target structure $\mathbf{r}$ and reconstruction $\hat{\mathbf{r}}$,

$$\mathcal{C}_{\mathrm{energy}}(\mathbf{r}_t, \hat{\mathbf{r}}_t) = \lambda(U(\mathbf{r_t}) - U(\hat{\mathbf{r}}_t))^2. \tag{8.5}$$

This term serves as a regularizer to improve the quality of backmapped structures. It accelerates convergence and helps to match the reconstructed energetics to the reference trajectory. Since the potential energy is sensitive to small perturbations of the coordinates, it can become dominatingly large during early stages of training before the model learns to localize atomic coordinates. As a remedy, the prefactor λ is incorporated, which is set to $\lambda = 0$ at the beginning of the training and slowly annealed up to $\lambda = 1$ using an exponential annealing schedule.

Finally, a regularization term $\mathcal{C}_{\mathrm{KL}}$ is applied to bias the approximate posterior $\hat{\mathbf{z}} = e\big(\mathcal{E}(\mathbf{r})\big)$ towards the desired prior distribution, i.e. a normal distribution $\mathcal{N}(0, \mathbf{I})$ [190],

$$\mathcal{C}_{\mathrm{KL}}(\hat{\mathbf{z}}) = \beta \mathcal{D}_{KL}(\hat{\mathbf{z}}||\mathcal{N}(0, \mathbf{I})), \tag{8.6}$$

where $\mathcal{D}_{KL}$ denotes the Kullback–Leibler (KL) divergence. The associated prefactor β scales the regularization loss and is set to $\beta = 1$ for the CLN model, while a cyclic annealing schedule is applied for ADP to mitigate vanishing of the KL term [254].

8.1.2 Markov State Model

Central to the evaluation of the proposed method is the analysis of dynamic properties. To this end, Markov state models (MSMs) are deployed to identify transitions between metastable states and their associated time scales [255, 256]. In particular, MSMs are a framework to analysis time-series data, which is often used for MD trajectories. At its core, an MSM decomposes configuration space into discrete and disjoint states, and describes the dynamics of the system by a transition matrix $\mathbf{P}$. Each element of the transition matrix $P_{ij}(\tau)$ denotes the transition probability from state i to state j during the lag time τ. In this work, first order MSMs are considered, which are memoryless models, i.e. transitions only depend on the current state. To construct MSMs from simulation data, the transition matrix $\mathbf{P}(\tau)$ is typically constructed in terms of collective variables $\mathbf{q}$, i.e. low-dimensional variables that characterize the configurational state of the system. Afterwards, the transition matrix $\mathbf{P}(\tau)$ can be decomposed into eigenvalues λ_i and eigenvectors Ψ_i,

$$\mathbf{P}(\tau)\Psi_i = \lambda_i \Psi_i. \tag{8.7}$$

In particular, the eigenvectors Ψ_i approximate the eigenfunctions of the transfer operator, i.e. the continuous integral operator that the transition probability matrix approximates [257]. Assuming that the system is in thermodynamic equilibrium, ergodic (any state can be reached starting from any other state given enough time) and aperiodic (different initializations of the system will lead to same equilibrium distribution), the following statements about eigenvalues λ_i and eigenvectors Ψ_i can be made [255]: (1) The elements of the eigenvectors correspond to each of the considered states. As such, the eigenvectors describe which states are contributing to the process identified by the associated eigenvalue. (2) The largest eigenvalue is always $\lambda_1 = 1$ and corresponds to the stationary distribution. (3) Subsequent eigenvalues $1 > \lambda_{i>1} > 0$ are associated with characteristic timescales, also called implied timescales, of dynamic processes described by the eigenvectors $\Psi_{i>1}$ that decay to equilibrium.

In this work, MSMs are used to compare dynamic properties of AA and CG simulations. Note that CG force fields typically yield faster simulation dynamics compared to AA force fields. Moreover, dynamics of the CG system are generally not accelerated by a constant factor, i.e. implied timescales of different processes can be rescaled non-uniformly. Therefore, ratios of implied timescales are used for a comparison of AA and CG dynamics.

8.2 Set-up and Reference Data

The proposed method is applied to the backmapping of two biomolecular systems: Alanine dipeptide (ADP) and the mini protein chignolin (CLN). Illustrations of both molecules at the AA and CG level can be found in Fig. 8.2. Data sets for training and testing of the model consist of pairs of corresponding AA and CG trajectories, which are obtained by mapping AA trajectories onto the CG representation. Since the test set is obtained similarly to the training set, it will be referred to as in-distribution test set in the following. Moreover, a generalization test set is constructed that consists of CG trajectories obtained with a MD simulation performed at the CG resolution. To this end, a CG force field is deployed that has been generated by CGSchNet, which is a ML-based method for force field parameterization [258, 251].

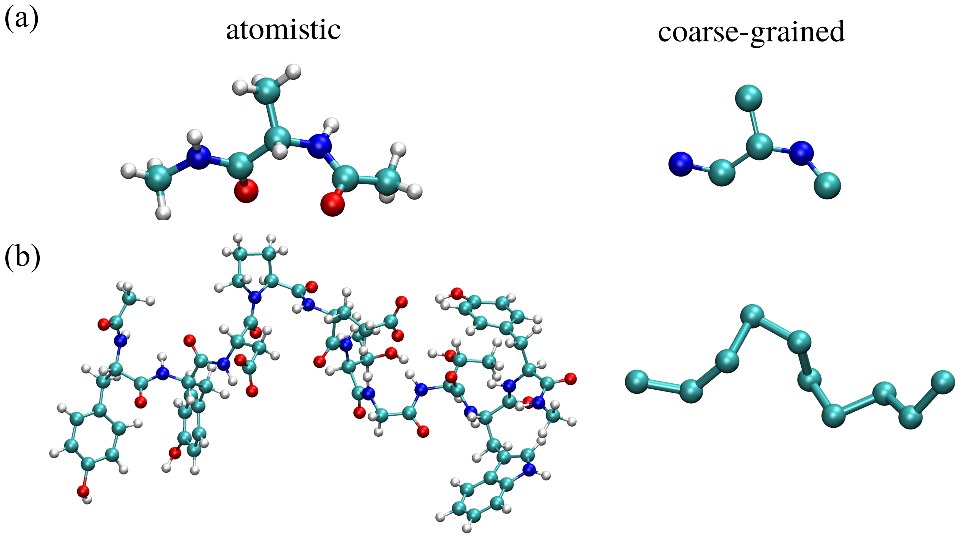

FIGURE 8.2: AA (left) and CG (right) representations of (a) alanine dipeptide and (b) chignolin.

8.2.1 Alanine Dipeptide

ADP mimics the dynamics of the amino acid alanine in a peptide chain and has been used as a model system in numerous previous studies [259, 260, 261, 262].

AA trajectories for ADP are obtained by MD simulations performed in explicit water. Simulations are carried out in the microcanonical (NVE) ensemble using the molecular dynamics package OpenMM [30]. The simulation procedure is based on the protocol outlined in [262]. In particular, the AMBER ff-99SB-ILDN force field is deployed and a cubic box containing 651 TIP3P water molecules randomly placed within a volume of (2.7273 nm)^3 is used [263]. The length of all bonds involving hydrogen atoms are constrained. A time step of 2.0 fs is used and initial velocities are sampled from a Maxwell-Boltzmann distribution at 300 K. During production, snapshots are recorded every 1.0 ps. The training set comprises 500000 and the test set 250000 frames, respectively.

The 22 atoms of ADP are coarse-grained into 6 beads. More specifically, the CG representation for ADP consists of 5 backbone carbon and nitrogen atoms and the beta carbon of the alanine side chain. Water molecules are treated implicitly, i.e. water is removed from the representation. CG forces obtained from the AA simulations are used for the training of CGSchNet and the training routine follows the procedure in [251]. The force field produced by CGSchNet is deployed to generate generalization data. The MD settings for the CG simulation are equivalent to the settings used for the AA simulation, except for an increased integration time step of 4.0 fs. Snapshots are recorded every 1.0 ps and a total of 400000 samples are collected.

8.2.2 Chignolin

The proposed method is also tested on a much more challenging data set of the mini protein CLN, which is composed of 10 amino acids plus termini. CLN displays a clear folding/unfolding transition when solvated in water [264].

Reference AA trajectories for CLN are provided by Wang *et al.* and are already reported in [258]. In particular, MD simulations are performed using the MD software ACEMD [265] deploying the CHARMM22* [266] force field and the TIP3P [267] water model. Simulations are carried out in the NVT ensemble at 350 K. Adaptive sampling is used to sufficiently sample folding/unfolding transitions of CLN facilitated by a MSM [268]. 3744 independent trajectories of 50 ns are recorded aggregating a total simulation time of $\sim$ 187 μs. Within each trajectory samples are spaced by 100 ps. The training set comprises 3650 trajectories and the test set 94 trajectories. For additional details regarding the CLN simulations the reader is referred to the work of Wang *et al.* [258].

While CLN consists of 175 atoms, it is coarse-grained into 10 beads. In particular, the CG representation for CLN consists of the 10 sequential α-carbons along the molecular backbone. Generalization data is generated by CG simulations performed with OpenMM in the NVT ensemble at 350 K. 1000 independent trajectories are generated starting from random configurations mapped from the AA trajectories. Each CG trajectory consists of 4000 frames spaced by 100 ps.

8.3 Results

The performance of the trained model is evaluated in terms of its capability to reproduce energetic, structural and dynamic properties of the AA reference system. The model is used to backmap in-distribution as well as generalization data. The in-distribution test set denotes projections of reference AA trajectories onto the CG resolution. While one part of this data set is used for training, the other part is used to evaluate the baseline accuracy of the backmapping model in the following. On the other hand, the generalization test set corresponds to CG simulation data. Note that the generalization data represents a more difficult backmapping exercise, as the model has to generalize to unseen simulated data generated by a different, approximate force field than the model was trained on. As such, the final error when performing inference of the generalization test set is a combination of the baseline reconstruction error of the backmapping model and the error in the approximate potential of mean force from the CGSchNet model.

8.3.1 Energetics

The potential energy distributions displayed in Fig. 8.3 serve as an indicator for the overall structural similarity between AA reference and backmapped structures. The energy distributions obtained for ADP shown in panel (a) reveal that the ML model is able to reproduce energetic properties with remarkable accuracy. While small high-energy tails can be observed for reconstructed molecules, the overall agreement of both test sets with the reference system is excellent.

Turning to the energy distributions for the more challenging mini protein CLN in panel (b) indicates a similar performance. However, the model suppresses structures with low energies compared to the reference system. Moreover, a discrepancy between the distributions obtained for the in-distribution and generalization test sets can be observed. In particular, the energy distribution for the generalization set displays a tail towards high energies that is not observed in the in-distribution test set.

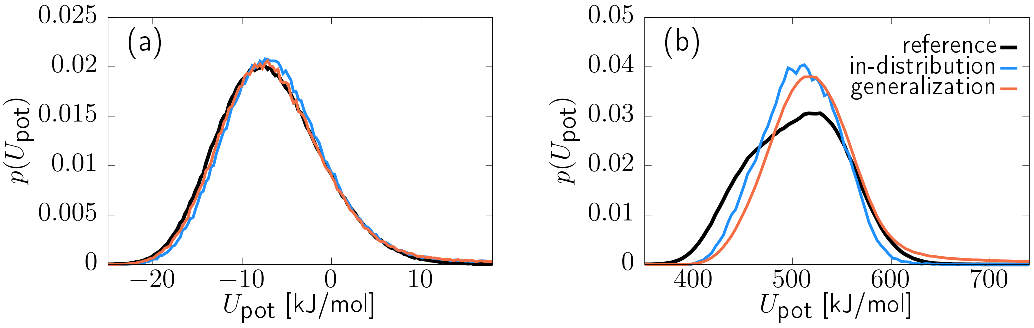

FIGURE 8.3: Distributions of the potential energy obtained for the reference system, backmapped in-distribution test set and backmapped generalization test set for (a) alanine dipeptide and (b) chignolin.

8.3.2 Free Energy Surfaces

In order to test structural agreement between the reference system and the backmapped test sets, free energy surfaces (FESs) are constructed. The FES are generated in the space of collective variables $\mathbf{q}$, i.e. low-dimensional variables that characterize the configurational state of the system. More specifically, relative populations $N(\mathbf{q}_i)$ are computed for discretized states $\mathbf{q}_i$ yielding free energies $F(\mathbf{q}_i) = -k_B T \ln(N(\mathbf{q})_\mathrm{i}) + \text{const.}$

FESs and selected snapshots for ADP can be found in Fig. 8.4. FESs are computed in terms of the backbone dihedrals ϕ and ψ, as they are well known collective variables to describe the conformational states of ADP [261, 260]. Panel (a) displays the FES obtained for the reference data. Three characteristic metastable states are observed that correspond to β-sheet (snapshots 1 and 2), α-helix (snapshot 3), and left-handed α-helix (snapshots 4 and 5) conformations of the amino acid. The baseline accuracy of the model is evaluated by analyzing AA reconstructions for the in-distribution test set, which can be found in panel (b). The model accurately reproduces all metastable states and is visually in excellent agreement with the reference FES. In addition, the model is transferred to the generalization test set, as shown in panel (c). The FES obtained for the backmapped generalization test set matches remarkably well with the reference FES. However, some regions along transition paths between metastable states display higher relative populations compared to the reference system, for example ($\phi \approx -2, \psi \approx -2$). While the CG force field enables broader and more frequent exploration of these regions of configuration space, they are underrepresented in the AA trajectory. Therefore, it is remarkable that the ML model generalizes to those sparsely sampled areas and reconstructs high-energy configurations accordingly. The structural fidelity of reconstructed configurations is further highlighted by superimposed collections of snapshots displayed in panel (d). Note that backmapped structures of a superposition are sampled using a fixed CG structure but varying latent samples $\mathbf{z}$. As such, the superpositions emphasize the non-deterministic aspect of the backmapping procedure. For both test sets, the ML model reconstructs visually faithful configurations with remarkable similarity to the AA reference data.

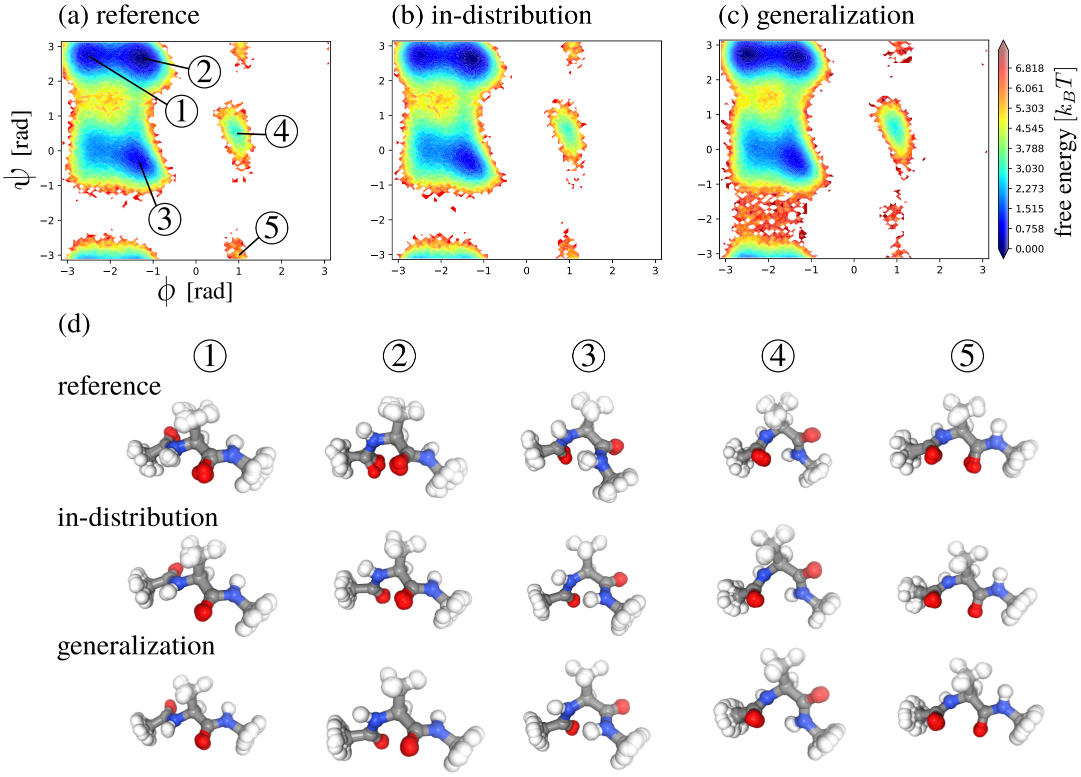

FIGURE 8.4: Comparison of atomistic and backmapped FES for ADP computed in terms of the backbone dihedrals Φ and Ψ. Ramanchandran plots are shown for (a) the atomistic reference in-distribution test set, (b) backmapped in-distribution test set and (c) backmapped generalization test set. Labels in (a) denote locations for the five metastable states of ADP. Panel (d) displays superimposed configurations for each metastable state

Fig. 8.5 displays the FESs and selected snapshots obtained for CLN. Unlike ADP, constructing meaningful collective variables for CLN is more challenging. To this end, time-lagged independent component analysis (TICA) is used for dimensionality reduction, as outlined in Sec. 3.2.3. The TICA algorithm is applied to the AA reference data to obtain a low-dimensional projection of the 45 pairwise α-carbon distances. In particular, the first two non-trivial independent components (ICs) are used as collective variables in the following. The FES obtained for the reference data is displayed in panel (a). Three metastable states can be identified that correspond to the folded state (snapshot 1), mis-folded state (snapshot 2) and unfolded state (snapshot 3). While all metastable states can be recovered upon backmapping of the in-distribution and generalization test sets (panel (b) and (c)), the FES for backmapped trajectories are contracted compared to the reference FES. In particular, the diversity of folded and mis-folded states is reduced upon backmapping compared to the reference system. This is also indicated by a lower variability of backmapped structures for the folded and mis-folded states displayed in panel (d). Similarly to ADP, backmapping of the generalization test set yields higher populations along the transition paths between metastable states compared to the in-distribution test set.

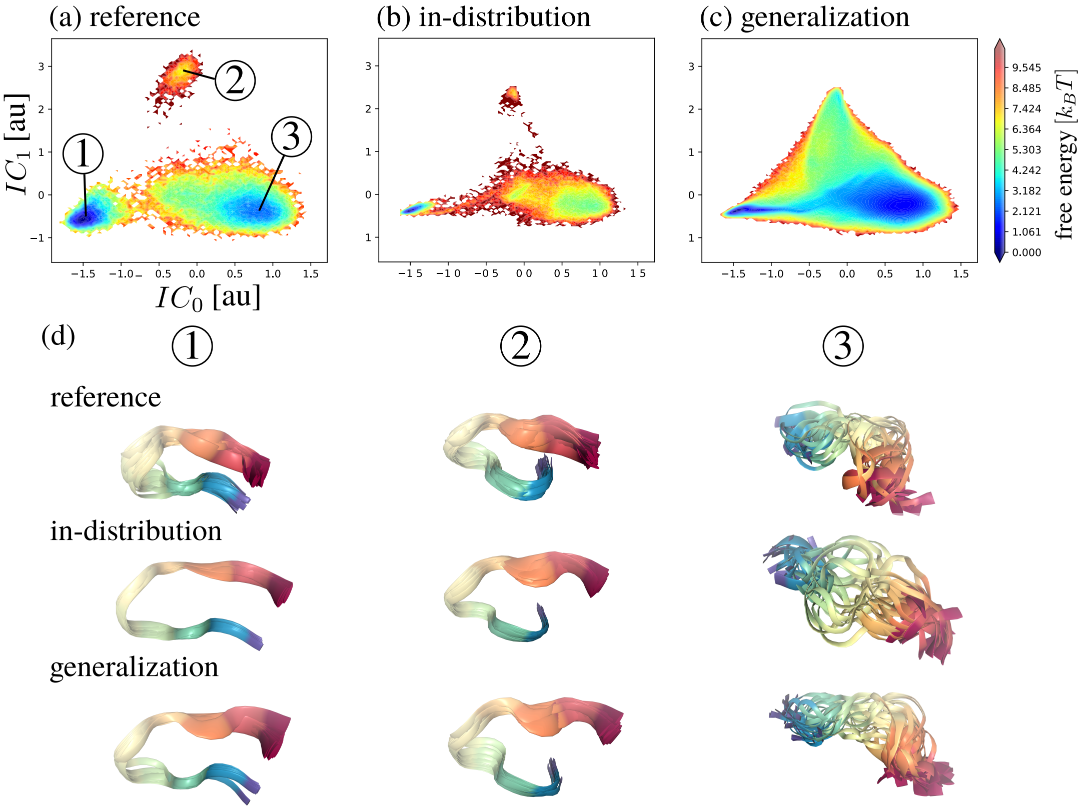

FIGURE 8.5: Comparison of atomistic and backmapped FES for CLN computed in terms of the first two non-trivial independent components obtained by time-lagged independent component analysis for the 45 pairwise α-carbon distances. Ramanchandran plots are shown for (a) the atomistic reference in-distribution test set, (b) backmapped in-distribution test set and (c) backmapped generalization test set. Labels in (a) denote locations for the three metastable states of CLN. Panel (d) displays superimposed configurations for each metastable state

8.3.3 Dynamics

A key feature of the proposed method is the incorporation of the previous trajectory frame as a conditional input for the ML model. Such temporal information is required to achieve temporal coherence between consecutive frames and sets the method apart from other backmapping schemes. In this section, kinetic properties of backmapped trajectories are analyzed in terms of implied timescales of slow processes obtained with MSMs. In addition, temporal coherence between frames is tested in terms of intra-frame velocities.

Timescales of Slow Processes

MSM are constructed as outlined in Sec. 8.1.2 deploying the collective variables used previously for the construction of FESs. In particular, the space of collective variables is decomposed using k-means clustering. For a direct comparison between timescales and processes between different MSMs, the same cluster centers obtained for the AA reference data are deployed for all data sets. To evaluate the similarity between processes, the cosine similarity c between two eigenvectors $\mathbf{\Psi}_i$ and $\mathbf{\Psi}_j$ is computed as

$$c = \frac{\mathbf{\Psi}_i \mathbf{\Psi}_j}{|\mathbf{\Psi}_i||\mathbf{\Psi}_j|}. \tag{8.8}$$

Moreover, collective variables for both systems can be computed at the CG resolution as well. Therefore, MSMs for CG trajectories prior to backmapping can be constructed additionally. Note that CG force fields typically yield faster simulation dynamics compared to AA force fields. To facilitate comparison between all trajectories, implied timescales obtained for CG simulation data are rescaled such that the timescales for the slowest process match.

Fig. 8.6 displays the implied timescales obtained for ADP trajectories. MSMs are built with a lag time of 5 ps and 100 cluster centers are used for the state decomposition. For all data sets, a cosine similarity > 90 % to the reference system is observed for the first three processes. A comparison of the implied timescales between AA reference and reconstructed in-distribution trajectories can be found in panel (a). While the first timescale matches by construction, the second and third timescales are also in excellent agreement and match within error. Note that subsequent timescales are below the resolution limit of the MSMs, since corresponding processes are faster than the applied lag time. Implied timescales for the first three processes obtained for the backmapped generalization set displayed in panel (b) also match remarkably well with the reference system. Moreover, timescales obtained for the backmapped generalization set and CG trajectories prior to backmapping are in excellent agreement. This indicates that the CG force field yields similar (but accelerated) dynamics compared to the AA force field and the ML model maintains the kinetics of slow motions present in the CG trajectories.

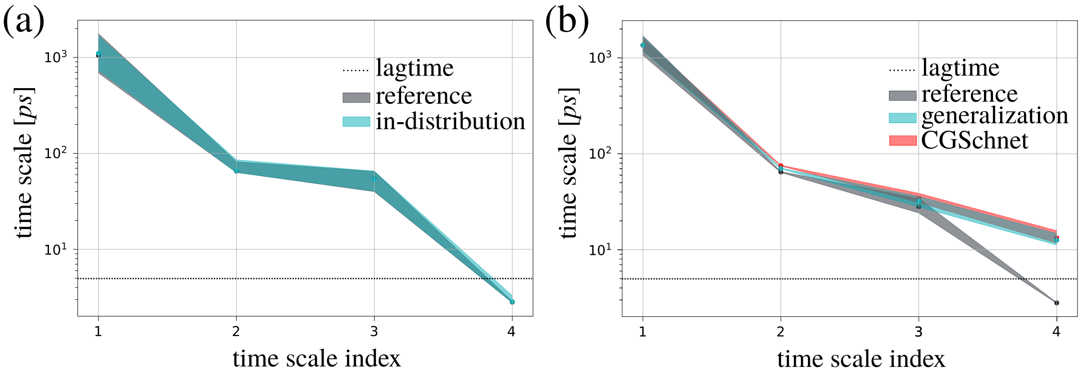

FIGURE 8.6: Comparison of implied timescales for ADP obtained by a MSM constructed in terms of the backbone dihedrals Φ and Ψ. Timescales are shown for (a) the atomistic reference in-distribution test set and backmapped in-distribution test set, as well as (b) the atomistic reference in-distribution test set, backmapped generalization test set and CG generalization test set.

A similar analysis for the timescales of slow processes obtained for CLN can be found in Fig. 8.7. The implied timescales obtained for the AA reference system are reproduced within error upon backmapping of the in-distribution test set, as can be seen in panel (a). However, a cosine similarity > 90 % is only observed for the first two processes, while the third process yields ≈ 80 % and the fourth process ≈ 60 % similarity. This indicates that the third and fourth slowest processes have slightly changed upon backmapping. Turning to the timescales obtained for the CGSchNet CG simulation in panel (b) reveals that timescales of different processes are not rescaled uniformly when the CG force field is deployed. While timescale ratios of the 1st, 3rd and 4th process are consistent with the kinetics observed for the AA reference system, the second process is accelerated more than the others. However, cosine similarities of the first and second process is ≈ 60 %, while a similarity < 25 % for the third and fourth process is observed. As such, timescale comparisons are not reliable, especially for the third and subsequent processes. Backmapping of the CG trajectory yields similar timescales compared to the CG kinetics for the first and second process, while the third and fourth process are slowed down. In conclusion, the CG force field for CLN yields dynamics that differ form the AA model and the backmapping method yields trajectories that reflect the slow dynamics of the underlying CG trajectories.

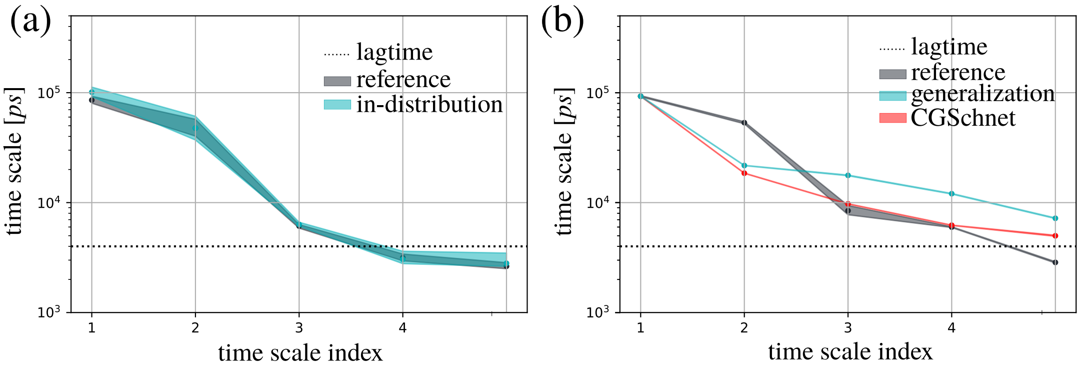

FIGURE 8.7: Comparison of implied timescales for CLN obtained by a MSM constructed in terms of the first two independent components of the time-lagged independent component analysis for the 45 pairwise α-carbon distances. Timescales are shown for (a) the atomistic reference in-distribution test set and backmapped in-distribution test set, as well as (b) the atomistic reference in-distribution test set, backmapped generalization test set and CG generalization test set.

Intra-frame Velocities

As a measure for temporal coherence, shifts of atomic positions between consecutive frames are analyzed, i.e. intra-frame velocities. In particular, atomic velocities $\mathbf{v}_i$ for a frame at time t are calculated as the deviations of atomic positions $\mathbf{s}_i$ between consecutive frames,

$$\mathbf{v}_i(t) = \frac{\mathbf{s}_i(t) - \mathbf{s}_i(t-\tau)}{\tau}, \tag{8.9}$$

where τ is the lag time between frames. Fig. 8.8 displays the intra-frame velocity distributions obtained for the reference trajectory and both reconstructed test sets. In addition, intra-frame velocity distributions obtained for a second backmapping method are shown, which is fragment-based and treats each frame separately. More specifically, a library consisting of 40000 pairs of equilibrated AA and associated CG frames is generated for both systems. Backmapping of a CG frame is performed by selecting the closest matching CG structure from the library in terms of root-mean-square deviation and projecting the corresponding AA structure onto the CG representation.

The velocity distributions for ADP can be found in panel (a). The backmapped distribution obtained for the in-distribution test set deploying the proposed temporal coherent backmapping scheme is in excellent agreement with the reference distribution. On the other hand, the frame-based method yields a velocity distribution for the in-distribution test set that differs from the reference system and is shifted towards slightly larger velocities. Backmapping of the generalization set deploying the ML model yields significantly larger intra-frame velocities, which is reasonable, as this reflects the acceleration of the CG dynamics.

The results obtained for CLN are displayed in panel (b). Temporal coherent backmapping yields a velocity distribution for the in-distribution test set that matches remarkably well with the reference distribution. Similarly to ADP, the frame-based method is not able to reproduce the reference velocities. Moreover, velocities obtained for the generalization set deploying the ML model are significantly larger compared to the reference system.

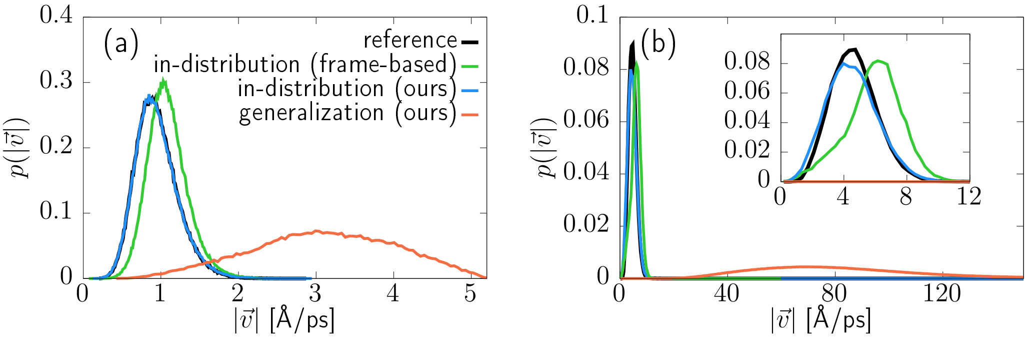

FIGURE 8.8: Comparison of the intra-frame velocity distributions for (a) alanine dipeptide and (b) chignolin as a measure for temporal coherence. Distributions are computed for the atomistic reference, backmapped in-distribution test set and backmapped generalization test set. In addition, a frame-based method is applied to the in-distribution test set.

8.4 Discussion

In this chapter, a new ML-based method for temporal coherent backmapping of molecular trajectories is introduced. In particular, a VAE is trained to non-deterministically reinsert atomistic details conditioned on the current CG and the previous AA frame.

The approach is applied to two popular biomolecular systems: ADP and the miniprotein CLN. The performance of the ML model is analyzed regarding its ability to reproduce energetic, structural and dynamic properties of the reference system. To evaluate the baseline accuracy of the model, the method is applied to an in-distribution test set that consists of AA structures projected onto the CG resolution. Excellent agreement between the reference system and the backmapped in-distribution test set is observed in terms of potential energy distributions. Moreover, structural properties match remarkably well, which is tested by analyzing FESs that are constructed in terms of collective variables. In order to analyze dynamic properties, MSMs are constructed in the space of collective variables to identify slow processes and their associated timescales. The obtained timescales of the backmapped trajectory agree remarkably well with the dynamics observed for the AA reference system. Temporal coherence between consecutive frames is evaluated in terms of intra-frame velocity distributions, which are reproduced with excellent accuracy deploying the ML model. The benefit of incorporating the previous state of the system as an additional input is highlighted by a comparison against a frame-based method, which yields velocity distributions that differ significantly from the reference.

In addition, backmapping of a generalization test set is performed, which consists of trajectories obtained in a CG simulation based on an approximate force field generated by CGSchNet. As such, this data represents a stress test for the generalizability of the backmapping model. Note that the final error is therefore a combination of the baseline reconstruction error of the backmapping model and the error in the approximate potential of mean force from the CGSchNet model. While energetic and structural properties of the reference system are reproduced with remarkable accuracy upon backmapping of this generalization data, dynamic properties differ. Timescale ratios of slow transitions between metastable states are recovered for ADP, but deviate from the AA reference for CLN. Moreover, intra-frame velocities of the backmapped generalization trajectories are significantly larger compared to the reference. However, a difference of dynamic properties is expected for this test set, since backmapped trajectories reflect the kinetics of the underlying CG trajectories rather than the AA reference system. CG simulations typically display faster dynamics compared to AA simulations as a direct consequence of deploying a reduced representation. Averaging over degrees of freedom effectively smoothens the energy landscape yielding a faster exploration of phase space. However, timescales for transitions between metastable states are typically not rescaled uniformly yielding timescale ratios that differ from the AA reference system [34].

Moreover, the VAE approach non-deterministically reinserts atomistic details along the CG variables. This feature is highlighted by a visual inspection of backmapped structures sampled from a fixed CG structure but differing latent samples. This procedure yields an ensemble of AA microstates that are all consistent with the given CG structure but still display variations.

In summary, the proposed method is able to backmap CG trajectories such that (1) each reconstructed frame has a high statistical weight, (2) each frame is a valid reconstruction of the given CG structure, i.e. atomistic degrees of freedom are reinserted along the CG variables and (3) consecutive frames are temporally coherent, i.e. shifts in atomic positions between consecutive frames follow the same distribution as the AA reference system. As such, the proposed method offers the ability to analyze the dynamics of a CG simulation at atomistic resolution.

Future work might focus on different strategies to improve the training protocol of the approach: (1) In order to improve sampling of configuration space, training samples of sparsely populated regions in configuration space can be emphasized more. This could be realized by accompanying training samples with thermodynamic or dynamical path weights. (2) An autoregressive training protocol could be applied to improve the temporal coherence. In particular, a recurrent neural network approach could add information of multiple consecutive frames to the gradients used during backpropagation. (3) To further encourage the ML model to utilize knowledge of previous states for its predictions, the training loss could be augmented with a reconstruction error of properties that are explicitly based on such information, for example intra-frame velocities.

Chapter 9

Conclusion and Outlook

To conclude this thesis, the main results of each chapter are summarized to highlight the importance of the results presented and the conclusions drawn. The first two sections recapitulate the underlying theory and refresh the motivation for this project. Afterwards, summaries of each of the subsequent chapters are presented, which restate the discussion sections of each chapter. Finally, an outlook for future research related to this thesis is given.

9.1 Multiscale Modeling

Computer simulations of molecular systems are routinely used to study molecular processes. The resolution of such computer models is generally only bound by computational effort. However, while quantum mechanics provides the most fundamental description of matter, ab initio molecular simulations quickly reach their limits. As a remedy, a coarser description of matter can be used to push the limits of accessible length- and timescales. In a first step, the resolution of molecular systems can be reduced to the level of single atoms. Such all-atom (AA) models are routinely implemented by molecular dynamics (MD) simulations that numerically integrate Newton's equation of motion. The interaction potentials for the atoms are often empirical and aim at correctly modeling structural, thermodynamic and/or dynamic properties of a target system [24]. However, the exploration of many relevant molecular processes, such as protein folding or binding, requires access to length- and timescales that are still out of reach for AA models [27, 28]. Therefore, the resolution is further reduced by averaging over atomistic degrees of freedom. Such coarse-grained (CG) models represent chemical compounds as particles in a similar fashion to AA MD simulations. In general, coarse-graining reduces the computational effort of the simulation and enables larger integration time steps [31, 32]. In addition, averaging over degrees of freedom yields "softer" interactions between coarse-grained sites and therefore, dynamics of the CG system are accelerated and a faster exploration of configuration space is possible.

The exploration of some phenomena require to consider wide range of length- and timescales, because molecular processes on multiple scales can be linked and interwoven. This is especially true for soft matter systems, where local interactions can impact large scale conformational changes. Therefore, a single model is sometimes not sufficient to capture the interplay of processes that are potentially linked to various different scales. As a solution, multiscale modeling (MM) can be applied, where models of different resolution are combined to address phenomena at multiple scales [38, 39, 36]: While coarse-graining is deployed to study the large-scale behavior of the system, higher resolution models are used to explore the behavior at local scales.

This thesis focuses on an important aspect of MM that is referred to as reverse-mapping: To establish a tight and consistent link between models of different resolutions, an approach to reintroduce details based on a lower resolution representation is required. Reverse-mapping is routinely used in the MM community to analyze simulation results on a local scale [40, 41, 42, 43], or for a direct comparison with experimental data, for example obtained with spectroscopic methods [44]. Moreover, reverse-mapping is applied to obtain a starting point for further high-resolution simulations [45, 41], or to asses the stability and accuracy of the obtained CG structures [45].

A reverse-mapping scheme has to generate new degrees of freedom and thereby take all their dependencies into account. In particular, generated microstates should be consistent with the given CG representation and should agree with the Boltzmann distribution at a particular state point. Most existing reverse-mapping schemes generate an initial atomistic structure that requires subsequent energy minimization for relaxation. In addition, MD simulations are typically performed to recover the correct statistical weights for the reinserted degrees of freedom. The computational effort for the subsequent energy minimization and equilibration procedures of such reverse-mapping schemes can become significant. As such, applications to large systems or high-throughput simulations are still limited. In addition, poorly initialized structures can get trapped into local minima with high energy barriers. Therefore, human intervention is frequently required that hinders the automation of such processes.

9.2 Machine Learning

In this work, machine learning (ML) is applied to improve the state-of-the-art in reverse-mapping of molecular structures. In the past decades, ML has emerged as a prominent research field that has a transformative impact on many domains, such as computer vision [6], speech recognition [7] or medical image analysis [8]. At its core, ML algorithms construct statistical models from data without relying on explicit program instructions. As such, the recent success of ML models is further fueled by the availability of large data sets. Recently, ML is gaining significant attention in many fields of modern science as well, especially particle physics and computational chemistry [9, 10, 11].

Within the field of ML, deep neural networks (DNNs) have received considerable attention. In particular, DNNs have dramatically improved the state-of-the-art in computer vision [6]. For example, deep generative models are able to synthesize photorealistic images of complex objects, such as human faces or animals [50, 51, 52]. At its core, DNNs are computational models that are based on a multiscale approach: Multiple layers are arranged subsequently and each layer transforms its input into a more abstract and composite representation, i.e. DNNs learn representations of data with multiple levels of abstraction. It is shown empirically that such deep learning approaches are successful in discovering complex structures in large data sets.

A milestone in the development of DNNs are convolutional neural networks (CNNs). Each layer of a CNN consists of a bank of convolutional kernels, also called filter, that slide over the input of the layer. This procedure yields a translation-equivariant response for the applied filters. Unlike traditional approaches, CNNs use parameterized filters, i.e. relevant pattern are learned from the data. This allows CNNs to learn a suitable representation of the data for a given task without relying on handcrafted features. In addition, the CNN approach benefits from weight sharing, i.e. learned filters are transferred across the whole input, which reduces the number of required parameters dramatically.

DNNs are routinely used for generative modeling, i.e. to provide an estimate for the probability $p_\Theta(\mathbf{x})$ of an observation $\mathbf{x}$, where Θ are the parameters of the model. The general goal is to approximate a target distribution $\mathcal{X}$, i.e. to find the optimal parameters Θ^* such that $p_{\Theta^*}(\mathbf{x}) \approx p_\mathcal{X}(\mathbf{x})$. The major route to train a generative model is to maximize the data likelihood. However, directly assessing the data likelihood is typically based on approximations or computational models that provide a tractable functional form for the likelihood, which might limit the expressivity of the model.

Implicit generative models do not require direct access of the likelihood function but define a stochastic procedure to generate new samples. Generative adversarial networks (GANs) have become one of the most successful implicit generative models known in the ML community [198, 199]. At its core, a GAN consists of two competing models trained in a game: A generator g_Θ produces synthetic samples by transforming samples $\mathbf{z}$ from a prior distribution. A second model, the discriminator c_Ψ with parameters Ψ, has to distinguish between synthetic samples $g_\Theta(\mathbf{z})$ from the generator and real samples $\mathbf{x}$ from the training set $\mathcal{D} = \{\mathbf{x}\}$, where $\mathbf{x}$ are drawn from the target distribution $\mathcal{X}$. As such, the discriminator c_Ψ acts as a distance measure for the target distribution $\mathcal{X}$ and the distribution of synthetic samples $g_\Theta(\mathcal{Z})$. This distance measure serves as a training objective for the generator g_Θ, i.e. g_Θ is trained to minimize this distance.

9.3 Methodology of Deepbackmap: Adversarial Reverse-mapping of Condensed-phase Molecular Structures

Chapter 4 forms the core of this thesis. The insights gained from the preceding theory chapters lead to the development of deepbackmap (DBM): A new method for the reverse-mapping of molecular structures based on ML. A key feature of DBM is its applicability to condensed-phase molecular systems. Unlike other reverse-mapping schemes, DBM aims at directly predicting equilibrated molecular structures that resemble the Boltzmann distribution. As such, the method does not rely on further energy minimization for relaxation and MD simulations for equilibration of the fine-grained structures. Moreover, DBM requires little human intervention, since the reinsertion of local details is learned from training examples.

DBM is trained with the generative adversarial approach. In particular, pairs of corresponding CG and fine-grained molecular structures are used for the training. While the fine-grained configurations serve as the target distribution, the CG structures are treated as conditional variables for the generative process: The generator has to generate missing degrees of freedom based on the CG structure. In order to evaluate the performance of the generator, a discriminative network is used to compare the generated structures with the training examples. Specifically, the input for the discriminator consists of both, the CG and the fine-grained configuration. As such, the discriminator evaluates not only the quality of the generated fine-grained structure, but also its consistency with the given CG structure.

The architecture of both models is based on CNNs. As the CNN architecture requires a regular discretization of 3D space, scaling to larger spatial structures is limited. Therefore, the generator is combined with an autoregressive approach that reconstructs the fine-grained structure incrementally, i.e. atom by atom. In addition, it is assumed that the placement of one atom relies only on short-range force field related features. In particular, DBM only learns local correlations while large-scale features are adapted from the CG structure. Therefore, only local information is required in each step, which makes the method scalable to larger system sizes. Moreover, it can be hypothesized that such local environments strongly overlap across different state points and across chemical space. As such, the local environment approach is a key feature for the generalizability of DBM.

Molecular graphs are generally undirected and can be cyclic or acyclic. For the graph traversal, which defines the order of reconstruction, the depth-first algorithm is used. In particular, the algorithm starts by sampling the variables with no parents from a prior distribution and generates subsequent variables based on the atoms generated in the previous steps. However, such forward sampling only yields accurate results if the underlying graph structure has a topological order, i.e. a graph traversal in which each node is visited only after all of its dependencies are explored. As such, accurate sampling of molecular structures calls for more feedback than a simple forward sampling strategy allows. This is especially true for condensed-phase systems, where great care has to be taken to avoid steric clashes. To this end, a variant of Gibbs sampling is applied, which subsequently refines the initial molecular structures by iteratively resampling the atom positions. Each further iteration still updates one atom at a time, but uses the knowledge of all other atoms.

Given the potential energy function of the system, the target distribution for the desired molecular structures is already known up to a normalization constant, i.e. the partition function. This knowledge can be incorporated in the training of DBM to improve its performance and to monitor the training process. Specifically, the potential energy U of generated structures is utilized as an additional term $\mathcal{C}_{\text{pot}}$ in the cost function of the generator. As such, $\mathcal{C}_{\text{pot}}$ acts as a regularizer that effectively narrows down the functional space of the generator by penalizing structures with high potential energy.

9.4 Performance and Transferability of DBM: Reverse-mapping of Syndiotactic Polystyrene

In chapter 5, the performance and transferability of the new reverse-mapping method DBM is evaluated. To this end, DBM is applied to a challenging condensed-phase molecular system of syndiotactic polystyrene (sPS). CG representations are obtained by a projection of AA data onto the CG resolution, where each sPS monomer is represented by two beads. In addition to DBM, a baseline backmapping method based on geometric rules and energy minimization (EM) is applied.

The general ability of DBM to reproduce a reference AA distribution from CG configurations is probed first. To this end, DBM is applied to high-temperature data of the polymeric system. DBM yields well-equilibrated configurations for this particular state point in terms of structural and energetic properties. The baseline method, on the other hand, over-stabilizes the system and therefore does not reproduce the specific state point accurately.

To probe the temperature transferability, training of DBM is fixed to melt configurations obtained at a high temperature and afterwards transferred to crystalline structures at lower temperatures. Again, DBM reproduces structural and energetic distributions of the reference system with remarkable accuracy. A higher-order investigation, facilitated by the Sketch-map (SM) algorithm, highlights the structural accuracy. The transferability of the baseline method to the crystalline phase is limited. In particular, MD simulations starting from backmapped configurations of the baseline method in the crystalline phase get stuck in local minima. Therefore, further human intervention would be required to achieve proper equilibration.

Finally, the chemical transferability of DBM is probed. To this end, training of DBM is fixed to liquids of octane and cumene. Afterwards, the model is transferred to the more complex sPS system without retraining. The performance of such chemically-transferred models varies in terms of bonded interactions: While the learned local correlations from octane and cumene allow for an accurate reconstruction of phenyl groups, reconstructed polymer backbones display discrepancies compared to the reference system. On the other hand, distributions of Lennard-Jones energies and pair correlation functions indicate that non-bonded features are reproduced with high accuracy. In addition, the SM algorithm is used to obtain a two-dimensional projection of the configuration space. It is observed that the reference and backmapped ensembles cover similar areas in this projected configuration space, but relative statistical weights of reference and backmapped microstates display discrepancies.

In summary, the ML-based method DBM is able to generate equilibrated AA molecular configurations based on CG structures. It is a well suited tool to automate backmapping processes as it learns the AA reconstruction from training data and therefore requires little human intervention. Moreover, avoiding unnecessary equilibrations of reverse-mapped structures will help to establish a tighter and more consistent link between models at different scales. In addition, DBM displays remarkable transferability features that can be linked to the applied local environment approach. In particular, the transferability across different state points can be rationalized in terms of a scale-separation: The backmapped structure is composed of the local correlations learned by DBM and the large-scale properties of the CG structure. It is hypothesized that most of the temperature dependence is carried by the CG structure. Local features on the other hand are assumed to be less temperature sensitive, because associated covalent interactions operate on energy scales significantly larger than $k_B T$. As such, local correlations separate from larger scales and therefore, can be transferred from the melt to the crystalline phase. Beside the transferability across state points, the advantages of the local environment approach of DBM are further highlighted by the encouraging performance of chemically-transferred models. In particular, it is demonstrated that small-scale features can be shared between different molecules, which allows DBM to interpolate across parts of chemical space. However, the limits of the generalization are shown as well by the limited quality of the reconstructed carbon backbone. It can be hypothesized that a bottleneck for the accuracy arises from missing features in the training set. Specifically, local environments of backbone carbons connecting monomers are not included in the training examples. As such, it can be expected that an increasing number of building blocks systematically improves the transferability across chemical space. In conclusion, DBM offers the perspective to recycle learned local correlations across different state points and across chemical space. This can be useful for future applications in MM, as DBM can be trained on data that is straightforward to obtain, such as liquids of small molecules or polymer melts with a small system size, but can be transferred to more challenging tasks afterwards, for example to study polymer crystallization or large systems of complex molecules.

9.5 Backmapping as a Quality Measure for Coarse-grained Models

In chapter 6, backmapping is applied to assess the quality of structure-based CG models at the AA resolution. In particular, three different models for Tris-Meta-Biphenyl-Triazine (TMBT) are parameterized that differ in their bonded interactions. While all CG models reproduce structural properties at the CG resolution targeted in the parameterization with remarkable accuracy, important cross-correlations between CG variables are not captured sufficiently. This is demonstrated by reverse-mapping of CG structures from two different sources: (1) An in-distribution test set denotes AA MD simulation data that is projected onto the CG resolution. One part of this data set is used to train DBM, while the other part is used to obtain the baseline accuracy of the backmapping method. (2) Snapshots obtained by MD simulations based on the CG force fields are denoted as generalization test sets. To assess the quality of the deployed CG models, backmapped in-distribution and backmapped generalization test sets are compared revealing significant discrepancies between the AA and CG ensembles.

DBM is able to reproduce AA pair correlation functions for the in-distribution test set with remarkable accuracy. However, application to the generalization test sets yields AA structures that contain steric clashes, i.e. non-bonded atoms that are too close to each other. To rationalize the deterioration in performance for the generalization test sets, consider the following requirements that a backmapping scheme has to fulfill: (1) Reconstructed AA details have to be consistent with the underlying CG structure and (2) backmapped structures have to agree with the Boltzmann distribution. It is demonstrated that the generalization test sets contain CG conformations that prohibit reconstructing AA details that fulfill both requirements simultaneously. In particular, DBM generates AA structures that are consistent with the CG structure but therefore contain unavoidable steric clashes. To underpin the results, a second method that relies on EM is applied for the backmapping. The EM-based method is more robust and displays a similar performance for both test sets. However, the EM-based method yields pair correlation functions that are overly peaked compared to the AA reference, which is reasonable because of the involved relaxation. More importantly, it is observed that the EM-based method generates structures with low potential energy but violates the consistency criteria, i.e. backmapped structures are shifted away from the underlying CG structure.

The reintroduced details enable force computations based on the AA force field. However, the coarse-to-fine mapping is not unique, as a single CG structure corresponds to an ensemble of AA microstates. Therefore, computed AA forces are projected onto the CG resolution to enable a more stringent comparison. The force distribution for the backmapped in-distribution test set obtained with DBM matches the AA reference distribution remarkably well. On the other hand, force distributions obtained for the generalization test sets display long tails towards large forces.

Computing the Jenson-Shannon (JS) divergence between force distributions of reference AA and backmapped configurations yields a clear ranking for the quality of the different CG models. On the other hand, force distributions obtained with the EM-based backmapping method are not insightful, since the involved relaxation yields indistinguishable force distributions that are shifted towards significantly smaller forces.

Assessing the quality of CG models at the AA resolution can be beneficial for the development of new CG force field parameterization strategies. For example, force distributions could be used to evaluate the CG ensemble in terms of the AA force field, such that CG configurations that yield large AA forces can be suppressed. An evident starting point for this strategy is the multiscale force-matching approach, where the parameters of the CG force field are tuned such that the CG potential approximates the average net AA forces. Note that the force-matching functional is evaluated in the AA ensemble, i.e. it only contains structural information regarding cross-correlations observed in the AA model. However, the CG model is in general not guaranteed to reproduce cross-correlations between different degrees of freedom perfectly. A force evaluation in terms of the CG ensemble can reveal inconsistencies of the cross-correlations and has therefore the potential to improve the force-matching strategy.

9.6 Morphing of Local Statistics: Mapping Through a Resolution Bottleneck

In chapter 7, DBM is applied to adjust local, structural properties of molecular configurations. The method aims at improving the quality of structures obtained with chemically-specific top-down models that capture the correct large-scale behavior of a target system, but yield less faithful representations on a local scale. In order to correct local discrepancies, molecular structures are projected onto a lower resolution, i.e. a resolution bottleneck, and DBM is used to reinsert degrees of freedom. Importantly, DBM is trained solely on structures of the target distribution. Afterwards, DBM is transferred to configurations obtained with the top-down model. As such, local details learned from the target distribution are inserted into the top-down structures, which is referred to as morphing of local properties.

The morphing approach is tested for Kuhn scale matched Kremer-Grest (KG) sPS melts. The sPS melts obtained with the KG model display similar large-scale properties, such as the mean square end-to-end distance, as the higher resolution and solely structure-based model by Fritz *et al.* However, Kuhn scale matching does not account for local properties below the Kuhn scale. As such, local structural distributions of both models differ, which is demonstrated by a projection of melt structures obtained with the Fritz model onto the resolution of the KG model. The impact of the morphing is evaluated in terms of structural distributions, pair correlation functions and free energy surfaces computed in SM coordinates. While morphing has no significant impact on large-scale characteristics, it is able to reconstruct local properties of the Fritz distribution with remarkable accuracy.

In addition, morphing of tetracosane (TCS) liquids obtained with the Martini model is performed. As a target system, AA simulations with the GROMOS-96 force field are used, which are projected onto the Martini resolution. While the morphing model is not able to correct local features sufficiently, the significant discrepancies between local properties of the Martini and the GROMOS configurations have to be emphasized. As such, it can be hypothesized that the distributions do not match at the resolution bottleneck.

The general goal of this project is to introduce a two-step backmapping scheme for top-down CG models. In a first step, local statistics of a CG structure are corrected before it is processed by a backmapping algorithm in a second step. To investigate the impact of the morphing on the quality of backmapped structures, DBM is trained to increase the resolution of sPS melts to the level of the original Fritz model and TCS liquids to the AA level. Only a minor impact of the morphing on the quality of backmapped structures is observed for both systems. In particular, backmapping of KG structures and CG Fritz structures already yield similar distributions of local structural features without morphing. This can be rationalized by the robust transferability of DBM: It can be hypothesized that strong local interactions at the higher resolution yield local correlations that separate from larger scales. Therefore, local correlations learned by DBM transfer well across the CG configuration space. As such, backmapping of KG structures yield similar local structural properties compared to backmapping of CG Fritz structures. On the other hand, small differences in the distributions of structural properties are observed between backmapped Martini and backmapped CG GROMOS TCS liquids. However, the morphing of TCS liquids has a limited accuracy, such that it is not able yo improve the quality of the backmapped liquids significantly. Future work to improve the morphing capability of the model can focus on a hierarchical approach, where local features are successively adjusted on multiple scales.

9.7 Temporal Coherent Backmapping of Molecular Trajectories

In chapter 8, a new ML-based method for temporal coherent backmapping of molecular trajectories is introduced. In particular, temporal coherent backmapping refers to reproducing shifts of atomic positions between consecutive frames that are comparable to the AA reference system. The proposed method aims at both, generating well-equilibrated molecular structures for each individual frame, while maintaining temporal coherence within a series of frames. To this end, a variational autoencoder (VAE) is trained to reinsert atomistic details conditioned on the current CG *and* the previous AA frame. The approach is applied to two popular biomolecular systems: Alanine dipeptide (ADP) and the miniprotein chignolin (CLN).

The baseline accuracy of the ML model is evaluated regarding its ability to reproduce structural and dynamic properties of reference AA trajectories. To this end, the method is applied to AA trajectories projected onto the CG resolution, which are referred to as in-distribution test set. Excellent structural similarity between the reference system and the backmapped in-distribution test set is observed in terms of potential energy distributions and free energy surfaces computed with respect to collective variables. In order to analyze the dynamics of backmapped trajectories, Markov state models are constructed to identify slow processes and their associated timescales. The implied timescales of the backmapped trajectories agree remarkably well with the dynamics observed for the AA reference system. To evaluate temporal coherence between consecutive frames, intra-frame velocity distributions are computed. The ML model is able to reproduce intra-frame velocities with excellent accuracy. The advantage of incorporating the previous AA simulation frame is highlighted by a comparison with a frame-based backmapping scheme, which yields velocity distributions that differ significantly from the reference.

In addition, the method is applied to analyze the dynamics of trajectories obtained in a CG simulation. To this end, CG simulations are performed based on approximate force fields generated by CGSchNet, which is a ML based method for CG force field parameterization. While structural properties of the reference AA system are reproduced with remarkable accuracy upon backmapping of the CGSchNet trajectories, dynamic properties differ. Timescale ratios of slow transitions between metastable states are recovered for ADP, but deviate from the AA reference for CLN. Moreover, intra-frame velocities of the backmapped CGSchNet trajectories are significantly larger compared to the AA reference. These findings are reasonable for CG simulation data, since averaging over degrees of freedom effectively smoothens the energy landscape and therefore enables a faster exploration of phase space. As such, CG simulations display faster dynamics compared to AA simulations. Moreover, timescales for transitions between metastable states are typically not rescaled uniformly yielding timescale ratios that differ from the AA reference system [34].

In summary, the proposed method is able of reverse-mapping CG trajectories such that (1) each reconstructed frame has a high statistical weight, (2) each frame is a valid reconstruction of the given CG structure, i.e. atomistic degrees of freedom are reinserted along the CG variables and (3) consecutive frames are temporally coherent. As such, a tool to analyze the dynamics of a CG simulation at AA resolution is proposed, which is of relevance for a thorough analysis of dynamic properties that require atomistic details, such as the dynamic structure factor [248, 249].

9.8 Outlook

Future work pertaining the integration of generative ML approaches into the MM framework can proceed along multiple avenues of research. A promising route is to apply DBM to hierarchical modeling, where a particular system is described by a nested sequence of CG models [269, 270, 48]. Starting from the lowest resolution, CG structures are successively backmapped to higher resolutions until the AA level is reached. While hierarchical modeling is used routinely, it requires a significant amount of human effort, as force fields for each level of resolution have to be parameterized. Therefore, DBM could be a great advantage for hierarchical modeling, as the ML-based model learns the reverse-mapping from training data and does not necessarily require force field parameterizations. As such, it can be used to automate the process.

Another direction for future research is to focus on the transferability of DBM, which is explored in Chpt. 5. Further insights into the limits of chemical transferability can be gained by systematically increasing the number of building blocks included in the training and analyzing their impact on the quality of unseen molecules. In general, a model trained on a large and diverse data set has the potential to provide a general-purpose backmapping tool for a wide range of chemical systems.

In addition, parameterization strategies for CG force fields could benefit from DBM. As outlined in Chpt. 6, a new force matching approach based on an evaluation of forces at the AA resolution can be used to improve the accuracy of cross-correlations between CG variables. A thorough investigation of the proposed method requires to implement an automated scheme that iteratively performs a CG simulation, backmaps the obtained CG configurations and updates the force field parameters taking the atomistic forces into account.

Moreover, future work to improve the methodology of DBM can focus on two aspects: (1) The conventional CNN architecture used for DBM in this work is not rotationally equivariant and therefore, the model has to learn rotational symmetries. Can rotational equivariant network architectures be used to improve the accuracy of DBM and reduce the computational effort for the backmapping task? (2) The order of reconstruction relies on a depth-first-search of the molecular graph. However, this might not be the best strategy for reconstruction. Can DBM therefore learn the order of reconstruction, such that artifacts upon backmapping can be reduced?

In conclusion, MM can benefit from generative ML models in various ways. It is firmly believed that a tighter integration of ML approaches into the research field of computational chemistry will lead to significant advances in the future.

Contributions

In this section, the individual contributions for each chapter are stated in detail.

Chapter 4 and 5:
The original idea of DBM was developed by Marc Stieffenhofer, Michael Wand and Tristan Bereau. The implementation of DBM was carried out by Marc Stieffenhofer. The simulation data for syndiotactic polysteryne was provided by Chan Liu. Simulations of the octane and cumene systems were performed by Marc Stieffenhofer. Data analysis was carried out by Marc Stieffenhofer. The papers were written by Marc Stieffenhofer with critical commentary from Tristan Bereau and Michael Wand.

Marc Stieffenhofer, Michael Wand, Tristan Bereau
Adversarial reverse mapping of equilibrated condensed-phase molecular structures
Machine Learning: Science and Technology, Volume 1, Number 4
DOI: 10.1088/2632-2153/abb6d4
© IOP Publishing Ltd, 2020

Marc Stieffenhofer, Tristan Bereau, Michael Wand
Adversarial reverse mapping of condensed-phase molecular structures: Chemical transferability
APL Materials 9, Volume 9, Number 3
DOI: 10.1063/5.0039102
© AIP Publishing LLC, 2021

Chapter 6:
This chapter summarizes insights obtained during the course of a collaboration with Christoph Scherer, Falk May and Denis Andrienko. The original idea to use backmapping as a quality measure for coarse-grained models was developed by Marc Stieffenhofer and Denis Andrienko. Christoph Scherer parameterized the coarse-grained force field for Tris-Meta-Biphenyl-Triazine and carried out all simulations. Energy-based backmapping was performed by Christoph Scherer, while backmapping with DBM was performed by Marc Stieffenhofer. Data analysis and interpretation of the results were carried out by Marc Stieffenhofer.

Chapter 7:
The original idea of the morphing approach was developed by Marc Stieffenhofer, Michael Wand and Tristan Bereau. Implementation of the approach was carried out by Marc Stieffenhofer. All simulations and data analysis were performed by Marc Stieffenhofer

Chapter 8:
The work stems from a collaboration of Marc Stieffenhofer, Kirill Shmilovich, Moritz Hoffmann and Nick Charron. The project originates from the long program *Machine Learning for Physics and the Physics of Learning* at the Institute for Pure & Applied Mathematics that was held from 09.04.19 to 12.08.19 at the University of California, Los Angeles. The original idea of temporal coherent backmapping was introduced by Marc Stieffenhofer and further developed by all collaborators. Kirill Shmilovich carried out the implementation of the approach advised by Marc Stieffenhofer and Moritz Hoffmann. All-atom simulations of alanine dipeptide were performed by Kirill Shmillovich and all-atom trajectories for chignolin were provided by Jiang Wang. Coarse-grained force fields were parameterized by Nick Charron and coarse-grained simulations were performed by Nick Charron. Data analysis was performed by Kirill Shmillovich. The manuscript was written by all four collaborators.

Kirill Shnilovich, Marc Stieffenhofer, Nick Charron, Moritz Hoffmann
Temporally coherent backmapping of molecular trajectories from coarse-grain to atomistic resolution
arXiv preprint:2205.05213
DOI: 10.48550/ARXIV.2205.05213
© Creative Commons Attribution 4.0 International

Appendix A

Appendix

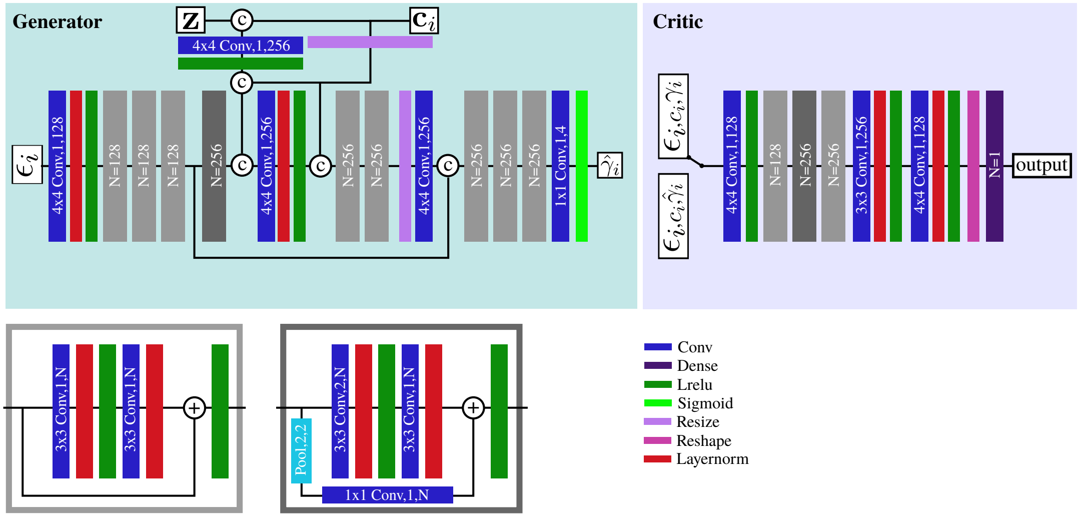

FIGURE A.1: CNN architecture with residual connections of the generator (left) and critic (right). The first part of the generator consists of an encoder which learns a lower dimensional embedding of the condition given by $\epsilon_i(\mathbf{x})$ using several residual connections and one pooling layer. Noise z and the atom type $\mathbf{c}_i$ are concatenated to this low dimensional embedding and is fed into the decoding part of the generator, which again consists of several residual connections and an upsampling layer. The critic learns a one dimensional embedding of the condition $\epsilon_i(\mathbf{x})$ and the target/fake atom $\gamma_i/\hat{\gamma}_i$ using residual layers and a final dense layer. Throughout the whole architecture layernorm is applied for regularization and LeakyRelus are used as nonlinearities.

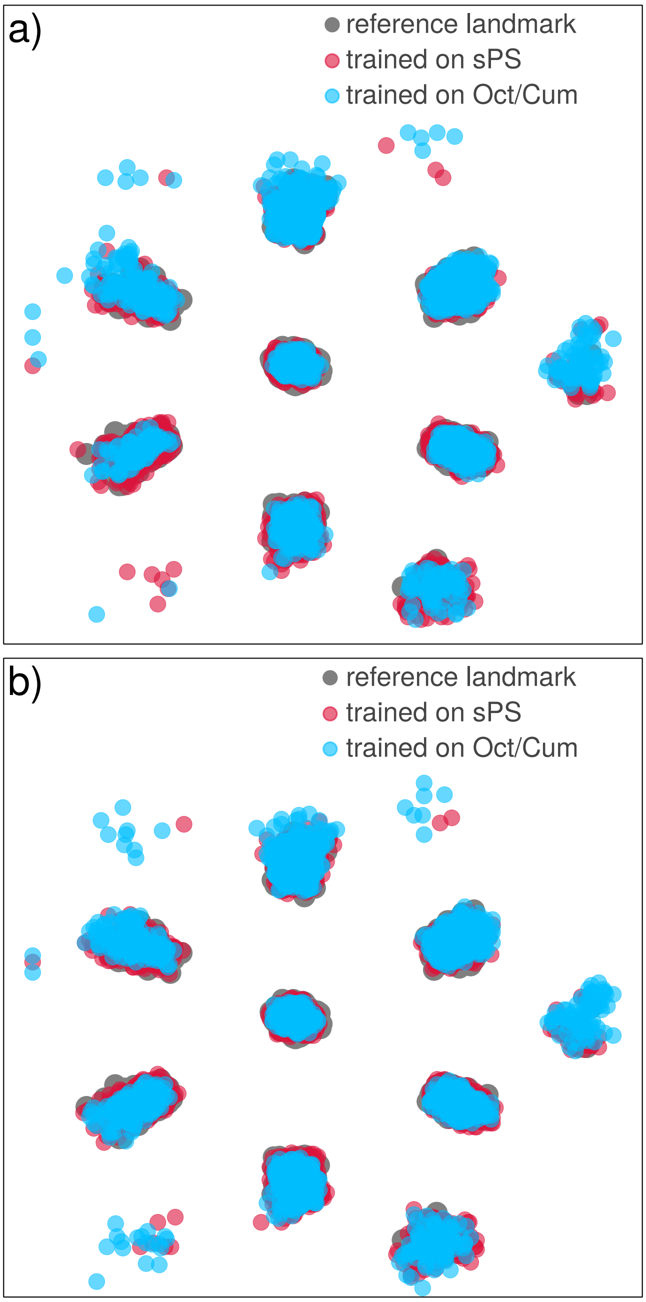

FIGURE A.2: Low-dimensional representation of the local environments of sPS monomers at $T = 568$ K. For each panel, snapshots are backmapped from identical CG configurations. Landmarks of reference structures (grey) and projections of structures generated with chemically-specific (red) and chemically-transferred (blue) models trained with (a) $\mathcal{C}^{(2)}_{\text{pot}}$ (b) without regularization. Reprinted from [223].

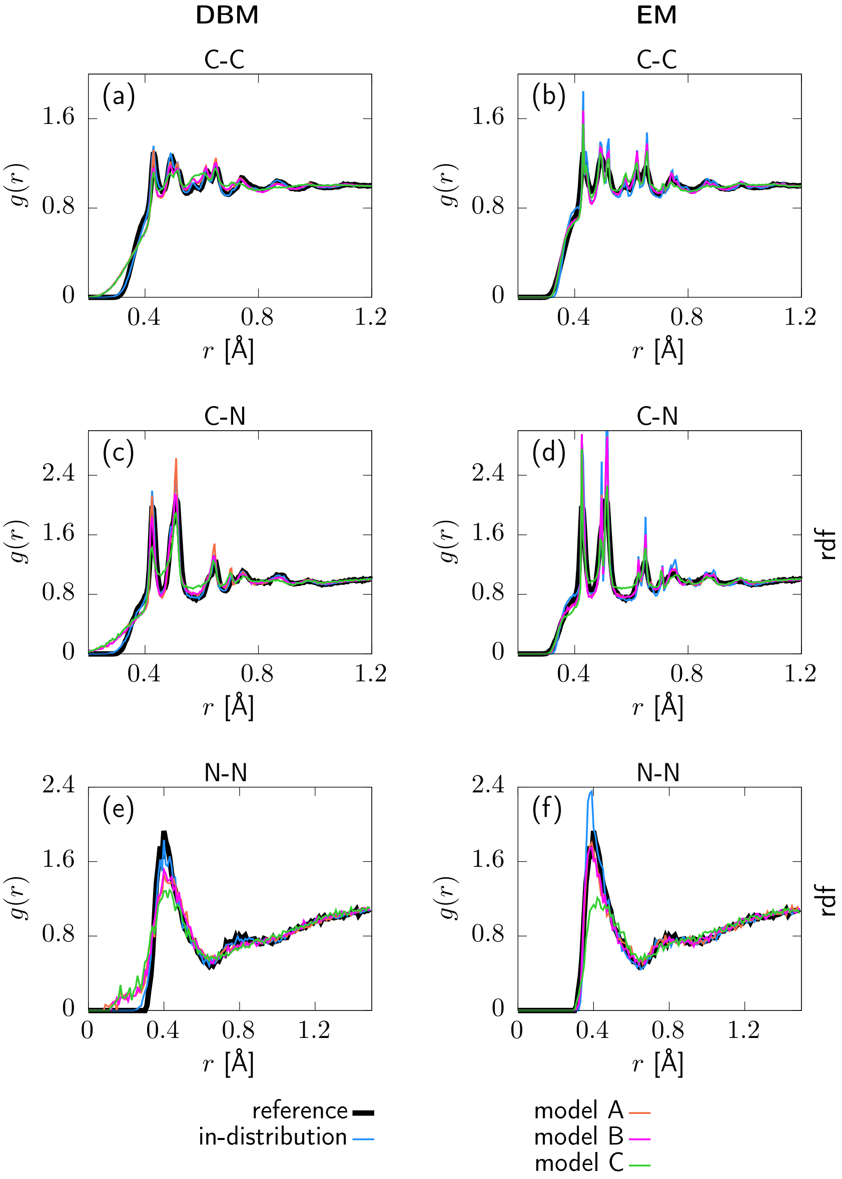

FIGURE A.3: Pair correlation functions $g(r)$ for the AA reference system, backmapped in-distribution test set and backmapped test set for all CG models. Results obtained with DBM (left) and EM scheme (right) are displayed, including non-bonded (a)-(b) C-C, (c)-(d) C-N and (e)-(f) N-N correlations.

Bibliography

[1] Michael Rubinstein, Ralph H Colby, et al. *Polymer physics*. Vol. 23. Oxford university press New York, 2003.

[2] Amy C Anderson. “The process of structure-based drug design”. In: *Chemistry & biology* 10.9 (2003), pp. 787–797.

[3] Christopher M Dobson. “Protein folding and misfolding”. In: *Nature* 426.6968 (2003), pp. 884–890.

[4] Errol Lewars. “Computational chemistry”. In: *Introduction to the theory and applications of molecular and quantum mechanics* (2011), p. 318.

[5] Andrew R Leach and Andrew R Leach. *Molecular modelling: principles and applications*. Pearson education, 2001.

[6] Athanasios Voulodimos et al. “Deep learning for computer vision: A brief review”. In: *Computational intelligence and neuroscience* 2018 (2018).

[7] Ali Bou Nassif et al. “Speech recognition using deep neural networks: A systematic review”. In: *IEEE access* 7 (2019), pp. 19143–19165.

[8] Meherwar Fatima, Maruf Pasha, et al. “Survey of machine learning algorithms for disease diagnostic”. In: *Journal of Intelligent Learning Systems and Applications* 9.01 (2017), p. 1.

[9] Frank Noé et al. “Machine learning for molecular simulation”. In: *Annual review of physical chemistry* 71 (2020), pp. 361–390.

[10] Jessica Vamathevan et al. “Applications of machine learning in drug discovery and development”. In: *Nature Reviews Drug Discovery* 18.6 (2019), pp. 463–477.

[11] Alexander Radovic et al. “Machine learning at the energy and intensity frontiers of particle physics”. In: *Nature* 560.7716 (2018), pp. 41–48.

[12] Jörg Behler. “Perspective: Machine learning potentials for atomistic simulations”. In: *The Journal of chemical physics* 145.17 (2016), p. 170901.

[13] Andreas Mardt et al. “VAMPnets for deep learning of molecular kinetics”. In: *Nature communications* 9.1 (2018), pp. 1–11.

[14] Marc Adrian et al. “Cryo-electron microscopy of viruses”. In: *Nature* 308.5954 (1984), pp. 32–36.

[15] Jean-Paul Renaud et al. "Cryo-EM in drug discovery: achievements, limitations and prospects". In: *Nature reviews Drug discovery* 17.7 (2018), pp. 471–492.

[16] John C Kendrew et al. "A three-dimensional model of the myoglobin molecule obtained by x-ray analysis". In: *Nature* 181.4610 (1958), pp. 662–666.

[17] Koichi Takahashi, Sorin Tănase-Nicola, and Pieter Rein Ten Wolde. "Spatio-temporal correlations can drastically change the response of a MAPK pathway". In: *Proceedings of the National Academy of Sciences* 107.6 (2010), pp. 2473–2478.

[18] Eric Betzig et al. "Imaging intracellular fluorescent proteins at nanometer resolution". In: *science* 313.5793 (2006), pp. 1642–1645.

[19] Bi-Chang Chen et al. "Lattice light-sheet microscopy: imaging molecules to embryos at high spatiotemporal resolution". In: *Science* 346.6208 (2014), p. 1257998.

[20] James M Holton. "A beginner's guide to radiation damage". In: *Journal of synchrotron radiation* 16.2 (2009), pp. 133–142.

[21] Krishnan Raghavachari. "Perspective on "Density functional thermochemistry. III. The role of exact exchange"". In: *Theoretical Chemistry Accounts* 103.3 (2000), pp. 361–363.

[22] Julian Tirado-Rives and William L Jorgensen. "Performance of B3LYP density functional methods for a large set of organic molecules". In: *Journal of chemical theory and computation* 4.2 (2008), pp. 297–306.

[23] Norbert Schuch and Frank Verstraete. "Computational complexity of interacting electrons and fundamental limitations of density functional theory". In: *Nature physics* 5.10 (2009), pp. 732–735.

[24] Alexander D MacKerell Jr. "Empirical force fields for biological macromolecules: overview and issues". In: *Journal of computational chemistry* 25.13 (2004), pp. 1584–1604.

[25] Tom Darden, Darrin York, and Lee Pedersen. "Particle mesh Ewald: An N log (N) method for Ewald sums in large systems". In: *The Journal of chemical physics* 98.12 (1993), pp. 10089–10092.

[26] Ben Leimkuhler and Charles Matthews. "Molecular dynamics". In: *Interdisciplinary applied mathematics* 36 (2015).

[27] Nuria Plattner and Frank Noé. "Protein conformational plasticity and complex ligand-binding kinetics explored by atomistic simulations and Markov models". In: *Nature communications* 6.1 (2015), pp. 1–10.

[28] Fabian Paul et al. “Protein-peptide association kinetics beyond the seconds timescale from atomistic simulations”. In: *Nature communications* 8.1 (2017), pp. 1–10.

[29] David E Shaw et al. “Anton 2: raising the bar for performance and programmability in a special-purpose molecular dynamics supercomputer”. In: *SC’14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE. 2014, pp. 41–53.

[30] Peter Eastman et al. “OpenMM 7: Rapid development of high performance algorithms for molecular dynamics”. In: *PLoS computational biology* 13.7 (2017), e1005659.

[31] Siewert J Marrink et al. “Comment on “On using a too large integration time step in molecular dynamics simulations of coarse-grained molecular models” by M. Winger, D. Trzesniak, R. Baron and WF van Gunsteren, Phys. Chem. Chem. Phys., 2009, 11, 1934”. In: *Physical Chemistry Chemical Physics* 12.9 (2010), pp. 2254–2256.

[32] Steve O Nielsen et al. “A coarse grain model for n-alkanes parameterized from surface tension data”. In: *The Journal of chemical physics* 119.14 (2003), pp. 7043–7049.

[33] Praveen K Depa and Janna K Maranas. “Speed up of dynamic observables in coarse-grained molecular-dynamics simulations of unentangled polymers”. In: *The Journal of chemical physics* 123.9 (2005), p. 094901.

[34] Dominik Fritz et al. “Multiscale modeling of soft matter: scaling of dynamics”. In: *Physical Chemistry Chemical Physics* 13.22 (2011), pp. 10412–10420.

[35] Matej Praprotnik, Luigi Delle Site, and Kurt Kremer. “Multiscale simulation of soft matter: From scale bridging to adaptive resolution”. In: *Annu. Rev. Phys. Chem.* 59 (2008), pp. 545–571.

[36] Christine Peter and Kurt Kremer. “Multiscale simulation of soft matter systems”. In: *Faraday discussions* 144 (2010), pp. 9–24.

[37] Christine Peter and Kurt Kremer. “Multiscale simulation of soft matter systems–from the atomistic to the coarse-grained level and back”. In: *Soft Matter* 5.22 (2009), pp. 4357–4366.

[38] Gary S Ayton, Will G Noid, and Gregory A Voth. “Multiscale modeling of biomolecular systems: in serial and in parallel”. In: *Current opinion in structural biology* 17.2 (2007), pp. 192–198.

[39] Gregory A Voth. *Coarse-graining of condensed phase and biomolecular systems*. CRC press, 2008.

[40] Pilar Brocos et al. “Multiscale molecular dynamics simulations of micelles: coarse-grain for self-assembly and atomic resolution for finer details”. In: *Soft Matter* 8.34 (2012), pp. 9005–9014.

[41] Yogendra Narayan Pandey et al. "Multiscale modeling of polyisoprene on graphite". In: *The Journal of chemical physics* 140.5 (2014), p. 054908.

[42] Sanket A Deshmukh et al. "Water ordering controls the dynamic equilibrium of micelle–fibre formation in self-assembly of peptide amphiphiles". In: *Nature communications* 7.1 (2016), pp. 1–11.

[43] Weria Pezeshkian et al. "Backmapping triangulated surfaces to coarse-grained membrane models". In: *Nature communications* 11.1 (2020), pp. 1–9.

[44] Berk Hess et al. "Long time atomistic polymer trajectories from coarse grained simulations: bisphenol-A polycarbonate". In: *Soft Matter* 2.5 (2006), pp. 409–414.

[45] Masahiro Shimizu and Shoji Takada. "Reconstruction of Atomistic Structures from Coarse-Grained Models for Protein–DNA Complexes". In: *Journal of chemical theory and computation* 14.3 (2018), pp. 1682–1694.

[46] Andrzej J Rzepiela et al. "Reconstruction of atomistic details from coarse-grained structures". In: *Journal of computational chemistry* 31.6 (2010), pp. 1333–1343.

[47] Tsjerk A Wassenaar et al. "Going backward: a flexible geometric approach to reverse transformation from coarse grained to atomistic models". In: *Journal of chemical theory and computation* 10.2 (2014), pp. 676–690.

[48] Guojie Zhang et al. "Hierarchical modelling of polystyrene melts: from soft blobs to atomistic resolution". In: *Soft Matter* 15.2 (2019), pp. 289–302.

[49] Antonio Brasiello, Silvestro Crescitelli, and Giuseppe Milano. "A multiscale approach to triglycerides simulations: from atomistic to coarse-grained models and back". In: *Faraday discussions* 158.1 (2012), pp. 479–492.

[50] Han Zhang et al. "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 5907–5915.

[51] Tero Karras et al. "Progressive growing of gans for improved quality, stability, and variation". In: *arXiv preprint arXiv:1710.10196* (2017).

[52] Andrew Brock, Jeff Donahue, and Karen Simonyan. "Large scale GAN training for high fidelity natural image synthesis". In: *arXiv preprint arXiv:1809.11096* (2018).

[53] Phillip Isola et al. "Image-to-image translation with conditional adversarial networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1125–1134.

[54] Jonathan Lahnsteiner et al. "Room-temperature dynamic correlation between methylammonium molecules in lead-iodine based perovskites: An ab initio molecular dynamics perspective". In: *Physical Review B* 94.21 (2016), p. 214114.

[55] Eric Paquet and Herna L Viktor. "Computational methods for Ab initio molecular dynamics". In: *Adv. Chem* 2018 (2018), p. 9839641.

[56] Jung Woo Lee et al. "Facile fabrication of sub-100 nm mesoscale inverse opal films and their application in dye-sensitized solar cell electrodes". In: *Scientific reports* 4.1 (2014), pp. 1–7.

[57] W Tschöp et al. "Simulation of polymer melts. I. Coarse-graining procedure for polycarbonates". In: *Acta Polymerica* 49.2-3 (1998), pp. 61–74.

[58] Siewert J Marrink et al. *The MARTINI force field*. CRC Press: New York, 2008.

[59] Elizabeth Villa, Alexander Balaeff, and Klaus Schulten. "Structural dynamics of the lac repressor–DNA complex revealed by a multiscale simulation". In: *Proceedings of the National Academy of Sciences* 102.19 (2005), pp. 6783–6788.

[60] Sergei Izvekov and Gregory A Voth. "A multiscale coarse-graining method for biomolecular systems". In: *The Journal of Physical Chemistry B* 109.7 (2005), pp. 2469–2473.

[61] Qiang Shi, Sergei Izvekov, and Gregory A Voth. "Mixed atomistic and coarse-grained molecular dynamics: simulation of a membrane-bound ion channel". In: *The Journal of Physical Chemistry B* 110.31 (2006), pp. 15045–15048.

[62] Matej Praprotnik, Luigi Delle Site, and Kurt Kremer. "Adaptive resolution molecular-dynamics simulation: Changing the degrees of freedom on the fly". In: *The Journal of chemical physics* 123.22 (2005), p. 224106.

[63] Matej Praprotnik, Luigi Delle Site, and Kurt Kremer. "A macromolecule in a solvent: Adaptive resolution molecular dynamics simulation". In: *The Journal of chemical physics* 126.13 (2007), 04B603.

[64] M Scott Shell. *Thermodynamics and statistical mechanics: an integrated approach*. Cambridge University Press, 2015.

[65] Donald A McQuarrie Simon and D John. *Physical chemistry: a molecular approach (Rev. ed.). Sausalito, Calif.: Univ*. 1997.

[66] Robert B Ash. *Information theory*. Courier Corporation, 2012.

[67] Claude Elwood Shannon. "A mathematical theory of communication". In: *The Bell system technical journal* 27.3 (1948), pp. 379–423.

[68] Edwin T Jaynes. "Information theory and statistical mechanics". In: *Physical review* 106.4 (1957), p. 620.

[69] Edwin T Jaynes. "Information theory and statistical mechanics. II". In: *Physical review* 108.2 (1957), p. 171.

[70] Michael P Allen et al. “Introduction to molecular dynamics simulation”. In: *Computational soft matter: from synthetic polymers to proteins* 23.1 (2004), pp. 1–28.

[71] Daan Frenkel and Berend Smit. *Understanding molecular simulation: from algorithms to applications*. Vol. 1. Elsevier, 2001.

[72] M Scott Shell. “Lecture notes in Principles of modern molecular simulation methods”. In: (2019).

[73] Dominik Marx and Jurg Hutter. “Ab initio molecular dynamics: Theory and implementation”. In: *Modern methods and algorithms of quantum chemistry* 1.301-449 (2000), p. 141.

[74] Justin A Lemkul et al. “An empirical polarizable force field based on the classical drude oscillator model: development history and recent applications”. In: *Chemical reviews* 116.9 (2016), pp. 4983–5013.

[75] Scott J Weiner et al. “A new force field for molecular mechanical simulation of nucleic acids and proteins”. In: *Journal of the American Chemical Society* 106.3 (1984), pp. 765–784.

[76] Bernard R Brooks et al. “CHARMM: a program for macromolecular energy, minimization, and dynamics calculations”. In: *Journal of computational chemistry* 4.2 (1983), pp. 187–217.

[77] William L Jorgensen, David S Maxwell, and Julian Tirado-Rives. “Development and testing of the OPLS all-atom force field on conformational energetics and properties of organic liquids”. In: *Journal of the American Chemical Society* 118.45 (1996), pp. 11225–11236.

[78] Philippe H. Hünenberger. “Thermostat Algorithms for Molecular Dynamics Simulations”. In: *Advanced Computer Simulation: Approaches for Soft Matter Sciences I*. Ed. by Christian Dr. Holm and Kurt Prof. Dr. Kremer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 105–149. ISBN: 978-3-540-31558-2. DOI: 10.1007/b99427. URL: https://doi.org/10.1007/b99427.

[79] Mark James Abraham et al. “GROMACS user manual version 5.0. 4”. In: *Sweden: Royal Institute of Technology and Uppsala University* (2014).

[80] Markus Deserno. “Mesoscopic membrane physics: concepts, simulations, and selected applications”. In: *Macromolecular rapid communications* 30.9-10 (2009), pp. 752–771.

[81] Michael Levitt and Arieh Warshel. “Computer simulation of protein folding”. In: *Nature* 253.5494 (1975), pp. 694–698.

[82] William George Noid. “Perspective: Coarse-grained models for biomolecular systems”. In: *The Journal of chemical physics* 139.9 (2013), 09B201_1.

[83] Maghesree Chakraborty, Chenliang Xu, and Andrew D White. "Encoding and selecting coarse-grain mapping operators with hierarchical graphs". In: *The Journal of chemical physics* 149.13 (2018), p. 134106.

[84] Marco Giulini et al. "An information-theory-based approach for optimal model reduction of biomolecules". In: *Journal of chemical theory and computation* 16.11 (2020), pp. 6795–6813.

[85] Adam Liwo et al. "Cumulant-based expressions for the multibody terms for the correlation between local and electrostatic interactions in the united-residue force field". In: *The Journal of Chemical Physics* 115.5 (2001), pp. 2323–2347.

[86] Reinier LC Akkermans and Willem J Briels. "A structure-based coarse-grained model for polymer melts". In: *The Journal of Chemical Physics* 114.2 (2001), pp. 1020–1031.

[87] Anthony J Clark et al. "Thermodynamic consistency in variable-level coarse graining of polymeric liquids". In: *Physical Review Letters* 109.16 (2012), p. 168301.

[88] WG Noid et al. "The multiscale coarse-graining method. I. A rigorous bridge between atomistic and coarse-grained models". In: *The Journal of chemical physics* 128.24 (2008), p. 244114.

[89] Clara D Christ, Alan E Mark, and Wilfred F Van Gunsteren. "Basic ingredients of free energy calculations: a review". In: *Journal of computational chemistry* 31.8 (2010), pp. 1569–1582.

[90] Raj Kumar Pathria. *Statistical mechanics*. Elsevier, 2016.

[91] Florian Müller-Plathe. "Coarse-graining in polymer simulation: From the atomistic to the mesoscopic scale and back". In: *ChemPhysChem* 3.9 (2002), pp. 754–769.

[92] Dirk Reith, Mathias Pütz, and Florian Müller-Plathe. "Deriving effective mesoscale potentials from atomistic simulations". In: *Journal of computational chemistry* 24.13 (2003), pp. 1624–1636.

[93] Furio Ercolessi and James B Adams. "Interatomic potentials from first-principles calculations: the force-matching method". In: *EPL (Europhysics Letters)* 26.8 (1994), p. 583.

[94] Sergei Izvekov and Gregory A Voth. "Multiscale coarse graining of liquid-state systems". In: *The Journal of chemical physics* 123.13 (2005), p. 134105.

[95] Solomon Kullback and Richard A Leibler. "On information and sufficiency". In: *The annals of mathematical statistics* 22.1 (1951), pp. 79–86.

[96] M Scott Shell. "The relative entropy is fundamental to multiscale and inverse thermodynamic problems". In: *The Journal of chemical physics* 129.14 (2008), p. 144108.

[97] Aviel Chaimovich and M Scott Shell. “Relative entropy as a universal metric for multiscale errors”. In: *Physical Review E* 81.6 (2010), p. 060104.

[98] Aviel Chaimovich and M Scott Shell. “Coarse-graining errors and numerical optimization using a relative entropy framework”. In: *The Journal of chemical physics* 134.9 (2011), p. 094112.

[99] Friederike Schmid. “Toy amphiphiles on the computer: What can we learn from generic models?” In: *Macromolecular rapid communications* 30.9-10 (2009), pp. 741–751.

[100] Cecilia Clementi. “Coarse-grained models of protein folding: toy models or predictive tools?” In: *Current opinion in structural biology* 18.1 (2008), pp. 10–15.

[101] François A Detcheverry et al. “Theoretically informed coarse grain simulations of block copolymer melts: method and applications”. In: *Soft Matter* 5.24 (2009), pp. 4858–4865.

[102] Chun Wu and Joan-Emma Shea. “Coarse-grained models for protein aggregation”. In: *Current opinion in structural biology* 21.2 (2011), pp. 209–220.

[103] John C Shelley et al. “Simulations of phospholipids using a coarse grain model”. In: *The Journal of Physical Chemistry B* 105.40 (2001), pp. 9785–9792.

[104] Wataru Shinoda, Russell DeVane, and Michael L Klein. “Multi-property fitting and parameterization of a coarse grained model for aqueous surfactants”. In: *Molecular Simulation* 33.1-2 (2007), pp. 27–36.

[105] Siewert J Marrink, Alex H De Vries, and Alan E Mark. “Coarse grained model for semiquantitative lipid simulations”. In: *The Journal of Physical Chemistry B* 108.2 (2004), pp. 750–760.

[106] JD Honeycutt and D Thirumalai. “Metastability of the folded states of globular proteins”. In: *Proceedings of the National Academy of Sciences* 87.9 (1990), pp. 3526–3529.

[107] Ira R Cooke, Kurt Kremer, and Markus Deserno. “Tunable generic model for fluid bilayer membranes”. In: *Physical Review E* 72.1 (2005), p. 011506.

[108] J-M Drouffe, AC Maggs, and S Leibler. “Computer simulations of self-assembled membranes”. In: *Science* 254.5036 (1991), pp. 1353–1356.

[109] Ken A Dill. “Theory for the folding and stability of globular proteins”. In: *Biochemistry* 24.6 (1985), pp. 1501–1509.

[110] Alena Shmygelska and Holger H Hoos. “An improved ant colony optimisation algorithm for the 2D HP protein folding problem”. In: *Conference of the Canadian Society for Computational Studies of Intelligence*. Springer. 2003, pp. 400–417.

[111] Kaizhi Yue et al. “A test of lattice protein folding algorithms”. In: *Proceedings of the National Academy of Sciences* 92.1 (1995), pp. 325–329.

[112] Kurt Kremer and Gary S Grest. "Dynamics of entangled linear polymer melts: A molecular-dynamics simulation". In: *The Journal of Chemical Physics* 92.8 (1990), pp. 5057–5086.

[113] Gary S Grest and Kurt Kremer. "Molecular dynamics simulation for polymers in the presence of a heat bath". In: *Physical Review A* 33.5 (1986), p. 3628.

[114] Roland Faller, Alexander Kolb, and Florian Müller-Plathe. "Local chain ordering in amorphous polymer melts: influence of chain stiffness". In: *Physical Chemistry Chemical Physics* 1.9 (1999), pp. 2071–2076.

[115] Ralf Everaers et al. "Kremer–grest models for commodity polymer melts: Linking theory, experiment, and simulation at the kuhn scale". In: *Macromolecules* 53.6 (2020), pp. 1901–1916.

[116] Carsten Svaneborg and Ralf Everaers. "Characteristic time and length scales in melts of Kremer–Grest bead–spring polymers with wormlike bending stiffness". In: *Macromolecules* 53.6 (2020), pp. 1917–1941.

[117] Siewert J Marrink et al. "The MARTINI force field: coarse grained model for biomolecular simulations". In: *The journal of physical chemistry B* 111.27 (2007), pp. 7812–7824.

[118] Luca Monticelli et al. "The MARTINI coarse-grained force field: extension to proteins". In: *Journal of chemical theory and computation* 4.5 (2008), pp. 819–834.

[119] Cesar A López et al. "Martini coarse-grained force field: extension to carbohydrates". In: *Journal of Chemical Theory and Computation* 5.12 (2009), pp. 3195–3210.

[120] Jaakko J Uusitalo et al. "Martini coarse-grained force field: extension to DNA". In: *Journal of chemical theory and computation* 11.8 (2015), pp. 3932–3945.

[121] Siewert J Marrink and D Peter Tieleman. "Perspective on the Martini model". In: *Chemical Society Reviews* 42.16 (2013), pp. 6801–6822.

[122] Riccardo Alessandri et al. "Pitfalls of the Martini model". In: *Journal of chemical theory and computation* 15.10 (2019), pp. 5448–5460.

[123] Christine Peter, Luigi Delle Site, and Kurt Kremer. "Classical simulations from the atomistic to the mesoscale and back: coarse graining an azobenzene liquid crystal". In: *Soft Matter* 4.4 (2008), pp. 859–869.

[124] Alessandra Villa, Christine Peter, and Nico FA van der Vegt. "Self-assembling dipeptides: conformational sampling in solvent-free coarse-grained simulation". In: *Physical Chemistry Chemical Physics* 11.12 (2009), pp. 2077–2086.

[125] Alessandra Villa, Nico FA van der Vegt, and Christine Peter. "Self-assembling dipeptides: including solvent degrees of freedom in a coarse-grained model". In: *Physical Chemistry Chemical Physics* 11.12 (2009), pp. 2068–2076.

[126] Wujie Wang and Rafael Gómez-Bombarelli. "Coarse-graining auto-encoders for molecular dynamics". In: *npj Computational Materials* 5.1 (2019), pp. 1–9.

[127] Wei Li et al. "Backmapping coarse-grained macromolecules: An efficient and versatile machine learning approach". In: *The Journal of Chemical Physics* 153.4 (2020), p. 041101.

[128] Yaxin An and Sanket A Deshmukh. "Machine learning approach for accurate backmapping of coarse-grained models to all-atom models". In: *Chemical Communications* 56.65 (2020), pp. 9312–9315.

[129] Johannes Hachmann et al. "The Harvard clean energy project: large-scale computational screening and design of organic photovoltaics on the world community grid". In: *The Journal of Physical Chemistry Letters* 2.17 (2011), pp. 2241–2251.

[130] Anubhav Jain et al. "Commentary: The Materials Project: A materials genome approach to accelerating materials innovation". In: *APL materials* 1.1 (2013), p. 011002.

[131] Camilo E Calderon et al. "The AFLOW standard for high-throughput materials science calculations". In: *Computational Materials Science* 108 (2015), pp. 233–238.

[132] Pankaj Mehta et al. "A high-bias, low-variance introduction to machine learning for physicists". In: *Physics reports* 810 (2019), pp. 1–124.

[133] David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowski. "A learning algorithm for Boltzmann machines". In: *Cognitive science* 9.1 (1985), pp. 147–169.

[134] Iqbal H Sarker. "Machine learning: Algorithms, real-world applications and research directions". In: *SN Computer Science* 2.3 (2021), pp. 1–21.

[135] Giuseppe Bonaccorso. *Machine learning algorithms*. Packt Publishing Ltd, 2017.

[136] Taiwo Oladipupo Ayodele. "Types of machine learning algorithms". In: *New advances in machine learning* 3 (2010), pp. 19–48.

[137] Sotiris B Kotsiantis, Ioannis Zaharakis, P Pintelas, et al. "Supervised machine learning: A review of classification techniques". In: *Emerging artificial intelligence applications in computer engineering* 160.1 (2007), pp. 3–24.

[138] Carlos Oscar Sánchez Sorzano, Javier Vargas, and A Pascual Montano. "A survey of dimensionality reduction techniques". In: *arXiv preprint arXiv:1403.2877* (2014).

[139] Amit Saxena et al. "A review of clustering techniques and developments". In: *Neurocomputing* 267 (2017), pp. 664–681.

[140] Lars Ruthotto and Eldad Haber. "An introduction to deep generative modeling". In: *GAMM-Mitteilungen* (2021), e202100008.

[141] Thomas Bayes. "LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFR S". In: *Philosophical transactions of the Royal Society of London* 53 (1763), pp. 370–418.

[142] Alan Hájek. "Interpretations of probability". In: (2002).

[143] Andrew Gelman et al. *Bayesian data analysis*. Chapman and Hall/CRC, 1995.

[144] Andrew Gelman et al. "Induction and deduction in Bayesian data analysis". In: *Rationality, Markets and Morals* 2.67-78 (2011), p. 1999.

[145] David L Donoho et al. "High-dimensional data analysis: The curses and blessings of dimensionality". In: *AMS math challenges lecture* 1.2000 (2000), p. 32.

[146] Michel Verleysen et al. "On the effects of dimensionality on data analysis with neural networks". In: *International Work-Conference on Artificial Neural Networks*. Springer. 2003, pp. 105–112.

[147] Charu C Aggarwal, Alexander Hinneburg, and Daniel A Keim. "On the surprising behavior of distance metrics in high dimensional space". In: *International conference on database theory*. Springer. 2001, pp. 420–434.

[148] Alexander N Gorban and Ivan Yu Tyukin. "Blessing of dimensionality: mathematical foundations of the statistical physics of data". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 376.2118 (2018), p. 20170237.

[149] Alexander N Gorban, Valery A Makarov, and Ivan Y Tyukin. "High-dimensional brain in a high-dimensional world: Blessing of dimensionality". In: *Entropy* 22.1 (2020), p. 82.

[150] Josiah Willard Gibbs. *Elementary principles in statistical mechanics: developed with especial reference to the rational foundation of thermodynamics*. Yale University Press, 1914.

[151] Laurens Van Der Maaten, Eric Postma, Jaap Van den Herik, et al. "Dimensionality reduction: a comparative". In: *J Mach Learn Res* 10.66-71 (2009), p. 13.

[152] Zhiqiang Ge. "Process data analytics via probabilistic latent variable models: A tutorial review". In: *Industrial & Engineering Chemistry Research* 57.38 (2018), pp. 12646–12661.

[153] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. "Testing the manifold hypothesis". In: *Journal of the American Mathematical Society* 29.4 (2016), pp. 983–1049.

[154] Jorge Lopez-Alvis et al. "Deep generative models in inversion: a review and development of a new approach based on a variational autoencoder". In: *arXiv preprint arXiv:2008.12056* (2020).

[155] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. "Multilayer feedforward networks are universal approximators". In: *Neural networks* 2.5 (1989), pp. 359–366.

[156] Gregory Naitzat, Andrey Zhitnikov, and Lek-Heng Lim. "Topology of Deep Neural Networks." In: *J. Mach. Learn. Res.* 21.184 (2020), pp. 1–40.

[157] Angel E García. "Large-amplitude nonlinear motions in proteins". In: *Physical review letters* 68.17 (1992), p. 2696.

[158] Rainer Hegger et al. "How complex is the dynamics of peptide folding?" In: *Physical review letters* 98.2 (2007), p. 028102.

[159] Mary A Rohrdanz et al. "Determination of reaction coordinates via locally scaled diffusion map". In: *The Journal of chemical physics* 134.12 (2011), 03B624.

[160] Michele Ceriotti, Gareth A Tribello, and Michele Parrinello. "Simplifying the representation of complex free-energy landscapes using sketch-map". In: *Proceedings of the National Academy of Sciences* 108.32 (2011), pp. 13023–13028.

[161] Gareth A Tribello, Michele Ceriotti, and Michele Parrinello. "Using sketch-map coordinates to analyze and bias molecular dynamics simulations". In: *Proceedings of the National Academy of Sciences* 109.14 (2012), pp. 5196–5201.

[162] Lutz Molgedey and Heinz Georg Schuster. "Separation of a mixture of independent signals using time delayed correlations". In: *Physical review letters* 72.23 (1994), p. 3634.

[163] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. "Independent component analysis, adaptive and learning systems for signal processing, communications, and control". In: *John Wiley & Sons, Inc* 1 (2001), pp. 11–14.

[164] Guillermo Pérez-Hernández et al. "Identification of slow molecular order parameters for Markov model construction". In: *The Journal of chemical physics* 139.1 (2013), 07B604_1.

[165] Christian R Schwantes and Vijay S Pande. "Improvements in Markov state model construction reveal many non-native interactions in the folding of NTL9". In: *Journal of chemical theory and computation* 9.4 (2013), pp. 2000–2009.

[166] Warren S McCulloch and Walter Pitts. "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5.4 (1943), pp. 115–133.

[167] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6 (1958), p. 386.

[168] Marvin Minsky and Seymour A Papert. *Perceptrons: An introduction to computational geometry*. MIT press, 2017.

[169] Jürgen Schmidhuber. "Deep learning in neural networks: An overview". In: *Neural networks* 61 (2015), pp. 85–117.

[170] Mark F Baer, Barry W Connors, and Michael A Paradiso. *Neurowissenschaften—Ein grundlegendes Lehrbuch für Biologie, Medizin und Psychologie [Neuroscience—A basic course book for biology, medicine, and psychology]*. 2009.

[171] Raúl Rojas. *Neural networks: a systematic introduction*. Springer Science & Business Media, 2013.

[172] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. "How does the brain solve visual object recognition?" In: *Neuron* 73.3 (2012), pp. 415–434.

[173] Zachary C Lipton, John Berkowitz, and Charles Elkan. "A critical review of recurrent neural networks for sequence learning". In: *arXiv preprint arXiv:1506.00019* (2015).

[174] George Cybenko. "Approximation by superpositions of a sigmoidal function". In: *Mathematics of Control, Signals and Systems* 5.4 (1992), pp. 455–455.

[175] Zhou Lu et al. "The expressive power of neural networks: A view from the width". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, pp. 6232–6240.

[176] Chiyuan Zhang et al. "Understanding deep learning (still) requires rethinking generalization". In: *Communications of the ACM* 64.3 (2021), pp. 107–115.

[177] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. "In search of the real inductive bias: On the role of implicit regularization in deep learning". In: *arXiv preprint arXiv:1412.6614* (2014).

[178] Hrushikesh Mhaskar, Qianli Liao, and Tomaso Poggio. "When and why are deep networks better than shallow ones?" In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1. 2017.

[179] Kenji Kawaguchi, Leslie Pack Kaelbling, and Yoshua Bengio. "Generalization in deep learning". In: *arXiv preprint arXiv:1710.05468* (2017).

[180] Behnam Neyshabur et al. "Exploring generalization in deep learning". In: *Advances in neural information processing systems* 30 (2017).

[181] Daniel Jakubovitz, Raja Giryes, and Miguel RD Rodrigues. "Generalization error in deep learning". In: *Compressed sensing and its applications*. Springer, 2019, pp. 153–193.

[182] Paul J Werbos. "Applications of advances in nonlinear sensitivity analysis". In: *System modeling and optimization*. Springer, 1982, pp. 762–770.

[183] Ian Goodfellow et al. "Generative adversarial networks". In: *Communications of the ACM* 63.11 (2020), pp. 139–144.

[184] Aaron van den Oord et al. "Conditional image generation with pixelcnn decoders". In: *arXiv preprint arXiv:1606.05328* (2016).

[185] Aaron van den Oord et al. "Wavenet: A generative model for raw audio". In: *arXiv preprint arXiv:1609.03499* (2016).

[186] Rafal Jozefowicz et al. “Exploring the limits of language modeling”. In: *arXiv preprint arXiv:1602.02410* (2016).

[187] Danilo Rezende and Shakir Mohamed. “Variational inference with normalizing flows”. In: *International conference on machine learning*. PMLR. 2015, pp. 1530–1538.

[188] Laurent Dinh, David Krueger, and Yoshua Bengio. “Nice: Non-linear independent components estimation”. In: *arXiv preprint arXiv:1410.8516* (2014).

[189] Ivan Kobyzev, Simon Prince, and Marcus Brubaker. “Normalizing flows: An introduction and review of current methods”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

[190] Diederik P Kingma and Max Welling. “Auto-encoding variational bayes”. In: *arXiv preprint arXiv:1312.6114* (2013).

[191] Diederik P Kingma and Max Welling. “An introduction to variational autoencoders”. In: *arXiv preprint arXiv:1906.02691* (2019).

[192] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. “Stochastic backpropagation and approximate inference in deep generative models”. In: *International conference on machine learning*. PMLR. 2014, pp. 1278–1286.

[193] Andrea Asperti, Davide Evangelista, and Elena Loli Piccolomini. “A Survey on Variational Autoencoders from a Green AI Perspective”. In: *SN Computer Science* 2.4 (2021), pp. 1–23.

[194] Scott E Fahlman, Geoffrey E Hinton, and Terrence J Sejnowski. “Massively parallel architectures for Al: NETL, Thistle, and Boltzmann machines”. In: *National Conference on Artificial Intelligence, AAAI*. 1983.

[195] Geoffrey E Hinton, Terrence J Sejnowski, et al. “Learning and relearning in Boltzmann machines”. In: *Parallel distributed processing: Explorations in the microstructure of cognition* 1.282-317 (1986), p. 2.

[196] Geoffrey E Hinton, Terrence J Sejnowski, and David H Ackley. *Boltzmann machines: Constraint satisfaction networks that learn*. Carnegie-Mellon University, Department of Computer Science Pittsburgh, PA, 1984.

[197] Ian Goodfellow et al. “Generative adversarial nets”. In: *Advances in neural information processing systems* 27 (2014).

[198] Antonia Creswell et al. “Generative adversarial networks: An overview”. In: *IEEE Signal Processing Magazine* 35.1 (2018), pp. 53–65.

[199] Jie Gui et al. “A review on generative adversarial networks: Algorithms, theory, and applications”. In: *IEEE Transactions on Knowledge and Data Engineering* (2021).

[200] Ian Goodfellow. “Nips 2016 tutorial: Generative adversarial networks”. In: *arXiv preprint arXiv:1701.00160* (2016).

[201] Tim Salimans et al. “Improved techniques for training gans”. In: *Advances in neural information processing systems* 29 (2016).

[202] Martin Arjovsky and Léon Bottou. “Towards principled methods for training generative adversarial networks”. In: *arXiv preprint arXiv:1701.04862* (2017).

[203] Sanjeev Arora et al. “Generalization and equilibrium in generative adversarial nets (gans)”. In: *International Conference on Machine Learning*. PMLR. 2017, pp. 224–232.

[204] Martin Arjovsky, Soumith Chintala, and Léon Bottou. “Wasserstein generative adversarial networks”. In: *International conference on machine learning*. PMLR. 2017, pp. 214–223.

[205] Jason D Lee et al. “Gradient descent only converges to minimizers”. In: *Conference on learning theory*. PMLR. 2016, pp. 1246–1257.

[206] Lucas Theis, Aäron van den Oord, and Matthias Bethge. “A note on the evaluation of generative models”. In: *arXiv preprint arXiv:1511.01844* (2015).

[207] Ishaan Gulrajani et al. “Improved training of wasserstein gans”. In: *Advances in neural information processing systems* 30 (2017).

[208] Marc Stieffenhofer, Michael Wand, and Tristan Bereau. “Adversarial reverse mapping of equilibrated condensed-phase molecular structures”. In: *Machine Learning: Science and Technology* 1.4 (2020), p. 045014.

[209] Laurianne David et al. “Molecular representations in AI-driven drug discovery: a review and practical guide”. In: *Journal of Cheminformatics* 12.1 (2020), pp. 1–22.

[210] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[211] Stuart Geman and Donald Geman. “Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images”. In: *IEEE Transactions on pattern analysis and machine intelligence* 6 (1984), pp. 721–741.

[212] Kunihiko Fukushima, Sei Miyake, and Takayuki Ito. “Neocognitron: A neural network model for a mechanism of visual pattern recognition”. In: *IEEE transactions on systems, man, and cybernetics* 5 (1983), pp. 826–834.

[213] Yann LeCun et al. “Backpropagation applied to handwritten zip code recognition”. In: *Neural computation* 1.4 (1989), pp. 541–551.

[214] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[215] Waseem Rawat and Zenghui Wang. “Deep convolutional neural networks for image classification: A comprehensive review”. In: *Neural computation* 29.9 (2017), pp. 2352–2449.

[216] Asifullah Khan et al. "A survey of the recent architectures of deep convolutional neural networks". In: *Artificial Intelligence Review* 53.8 (2020), pp. 5455–5516.

[217] Zhirong Wu et al. "3d shapenets: A deep representation for volumetric shapes". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1912–1920.

[218] Taco S Cohen and Max Welling. "Steerable cnns". In: *arXiv preprint arXiv:1612.08498* (2016).

[219] Tian Xie and Jeffrey C Grossman. "Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties". In: *Physical review letters* 120.14 (2018), p. 145301.

[220] Nathaniel Thomas et al. "Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds". In: *arXiv preprint arXiv:1802.08219* (2018).

[221] Karol Kurach et al. "The gan landscape: Losses, architectures, regularization, and normalization". In: (2018).

[222] Minhyeok Lee and Junhee Seok. "Regularization methods for generative adversarial networks: An overview of recent studies". In: *arXiv preprint arXiv:2005.09165* (2020).

[223] Marc Stieffenhofer, Tristan Bereau, and Michael Wand. "Adversarial reverse mapping of condensed-phase molecular structures: Chemical transferability". In: *APL Materials* 9.3 (2021), p. 031107.

[224] Chan Liu, Kurt Kremer, and Tristan Bereau. "Polymorphism of syndiotactic polystyrene crystals from multiscale simulations". In: *Advanced Theory and Simulations* 1.7 (2018), p. 1800024.

[225] Florian Müller-Plathe. "Local structure and dynamics in solvent-swollen polymers". In: *Macromolecules* 29.13 (1996), pp. 4782–4791.

[226] Berk Hess et al. "GROMACS 4: algorithms for highly efficient, load-balanced, and scalable molecular simulation". In: *Journal of chemical theory and computation* 4.3 (2008), pp. 435–447.

[227] Dominik Fritz et al. "Coarse-grained polymer melts based on isolated atomistic chains: Simulation of polystyrene of different tacticities". In: *Macromolecules* 42.19 (2009), pp. 7579–7588.

[228] Alpeshkumar K Malde et al. "An automated force field topology builder (ATB) and repository: version 1.0". In: *Journal of chemical theory and computation* 7.12 (2011), pp. 4026–4037.

[229] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[230] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[231] Albert P Bartók, Risi Kondor, and Gábor Csányi. “On representing chemical environments”. In: *Physical Review B* 87.18 (2013), p. 184115.

[232] Sandip De et al. “Comparing molecules and solids across structural and alchemical space”. In: *Physical Chemistry Chemical Physics* 18.20 (2016), pp. 13754–13769.

[233] Teemu Murtola et al. “Multiscale modeling of emergent materials: biological and soft matter”. In: *Physical Chemistry Chemical Physics* 11.12 (2009), pp. 1869–1892.

[234] Peter Májek and Ron Elber. “A coarse-grained potential for fold recognition and molecular dynamics simulations of proteins”. In: *Proteins: Structure, Function, and Bioinformatics* 76.4 (2009), pp. 822–836.

[235] Anirban Mondal et al. “Molecular library of OLED host materials—Evaluating the multiscale simulation workflow”. In: *Chemical Physics Reviews* 2.3 (2021), p. 031304.

[236] Victor Ruhle et al. “Versatile object-oriented toolkit for coarse-graining applications”. In: *Journal of Chemical Theory and Computation* 5.12 (2009), pp. 3211–3223.

[237] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. “Cross-modal retrieval with correspondence autoencoder”. In: *Proceedings of the 22nd ACM international conference on Multimedia*. 2014, pp. 7–16.

[238] Yue Cao et al. “Correlation autoencoder hashing for supervised cross-modal search”. In: *Proceedings of the 2016 ACM on international conference on multimedia retrieval*. 2016, pp. 197–204.

[239] Tanmoy Mukherjee, Makoto Yamada, and Timothy M Hospedales. “Deep matching autoencoders”. In: *arXiv preprint arXiv:1711.06047* (2017).

[240] Jaechang Yoo, Heesong Eom, and Yong Suk Choi. “Image-to-image translation using a cross-domain auto-encoder and decoder”. In: *Applied Sciences* 9.22 (2019), p. 4780.

[241] Kaiye Wang et al. “A comprehensive survey on cross-modal retrieval”. In: *arXiv preprint arXiv:1607.06215* (2016).

[242] Wilfred F van Gunsteren et al. “Biomolecular simulation: the GROMOS96 manual and user guide”. In: *Vdf Hochschulverlag AG an der ETH Zürich, Zürich* 86 (1996), pp. 1–1044.

[243] Jonathan D Halverson et al. “ESPResSo++: A modern multiscale simulation package for soft matter systems”. In: *Computer Physics Communications* 184.4 (2013), pp. 1129–1149.

[244] John P Bergsma et al. “Electronic spectra from molecular dynamics: a simple approach”. In: *The Journal of Physical Chemistry* 88.3 (1984), pp. 612–619.

[245] Bertrand Guillot. “A molecular dynamics study of the far infrared spectrum of liquid water”. In: *The Journal of chemical physics* 95.3 (1991), pp. 1543–1551.

[246] J. J. Salacuse, A. R. Denton, and P. A. Egelstaff. “Finite-size effects in molecular dynamics simulations: Static structure factor and compressibility. I. Theoretical method”. In: *Phys. Rev. E* 53 (3 1996), pp. 2382–2389. DOI: 10.1103/PhysRevE.53.2382. URL: https://link.aps.org/doi/10.1103/PhysRevE.53.2382.

[247] Neil E Moe and MD Ediger. “Calculation of the coherent dynamic structure factor of polyisoprene from molecular dynamics simulations”. In: *Physical Review E* 59.1 (1999), p. 623.

[248] Chunxia Chen et al. “Comparison of explicit atom, united atom, and coarse-grained simulations of poly (methyl methacrylate)”. In: *The Journal of chemical physics* 128.12 (2008), p. 124906.

[249] Arantxa Arbe, Fernando Alvarez, and Juan Colmenero. “Neutron scattering and molecular dynamics simulations: Synergetic tools to unravel structure and dynamics in polymers”. In: *Soft Matter* 8.32 (2012), pp. 8257–8270.

[250] Wujie Wang et al. “Generative Coarse-Graining of Molecular Conformations”. In: *arXiv preprint arXiv:2201.12176* (2022).

[251] Brooke E Husic et al. “Coarse graining molecular dynamics with graph neural networks”. In: *The Journal of chemical physics* 153.19 (2020), p. 194101.

[252] Kirill Shmilovich et al. “Temporally coherent backmapping of molecular trajectories from coarse-grain to atomistic resolution”. In: *arXiv preprint arXiv:2205.05213* (2022).

[253] Partha Ghosh et al. “From variational to deterministic autoencoders”. In: *arXiv preprint arXiv:1903.12436* (2019).

[254] Hao Fu et al. “Cyclical annealing schedule: A simple approach to mitigating kl vanishing”. In: *arXiv preprint arXiv:1903.10145* (2019).

[255] Brooke E Husic and Vijay S Pande. “Markov state models: From an art to a science”. In: *Journal of the American Chemical Society* 140.7 (2018), pp. 2386–2396.

[256] Frank Noé and Edina Rosta. *Markov models of molecular kinetics*. 2019.

[257] Ch Schütte et al. “A direct approach to conformational dynamics based on hybrid Monte Carlo”. In: *Journal of Computational Physics* 151.1 (1999), pp. 146–168.

[258] Jiang Wang et al. “Machine learning of coarse-grained molecular dynamics force fields”. In: *ACS central science* 5.5 (2019), pp. 755–767.

[259] Paul E Smith. "The alanine dipeptide free energy surface in solution". In: *The Journal of chemical physics* 111.12 (1999), pp. 5568–5579.

[260] Francesca Vitalini et al. "Dynamic properties of force fields". In: *The Journal of Chemical Physics* 142.8 (2015), 02B611_1.

[261] Feliks Nuske et al. "Variational approach to molecular kinetics". In: *Journal of chemical theory and computation* 10.4 (2014), pp. 1739–1752.

[262] Feliks Nüske et al. "Markov state models from short non-equilibrium simulations—Analysis and correction of estimation bias". In: *The Journal of Chemical Physics* 146.9 (2017), p. 094104.

[263] Kresten Lindorff-Larsen et al. "Improved side-chain torsion potentials for the Amber ff99SB protein force field". In: *Proteins: Structure, Function, and Bioinformatics* 78.8 (2010), pp. 1950–1958.

[264] Daisuke Satoh et al. "Folding free-energy landscape of a 10-residue miniprotein, chignolin". In: *FEBS letters* 580.14 (2006), pp. 3422–3426.

[265] Matt J Harvey, Giovanni Giupponi, and G De Fabritiis. "ACEMD: accelerating biomolecular dynamics in the microsecond time scale". In: *Journal of chemical theory and computation* 5.6 (2009), pp. 1632–1639.

[266] Stefano Piana, Kresten Lindorff-Larsen, and David E Shaw. "How robust are protein folding simulations with respect to force field parameterization?" In: *Biophysical journal* 100.9 (2011), pp. L47–L49.

[267] William L Jorgensen et al. "Comparison of simple potential functions for simulating liquid water". In: *The Journal of chemical physics* 79.2 (1983), pp. 926–935.

[268] Jan-Hendrik Prinz et al. "Markov models of molecular kinetics: Generation and validation". In: *The Journal of chemical physics* 134.17 (2011), p. 174105.

[269] Dominik Fritz et al. "Hierarchical modeling of polymer permeation". In: *Soft Matter* 5.22 (2009), pp. 4556–4563.

[270] Guojie Zhang et al. "Communication: One size fits all: Equilibrating chemically different polymer liquids through universal long-wavelength description". In: *The Journal of chemical physics* 142.22 (2015), p. 221102.