
Coarse-graining and inverse design in soft matter via local density-dependent potentials and machine learning methods

Fabian Berressem

geb. am 27.01.1994 in Lahnstein

Dissertation zur Erlangung des Grades
“Doktor der Naturwissenschaften”
am Fachbereich Physik, Mathematik und Informatik
der Johannes Gutenberg-Universität Mainz



JOHANNES GUTENBERG
UNIVERSITÄT MAINZ

First Supervisor: Dr. Arash Nikoubashman
Second Supervisor: Prof. Dr. Friederike Schmid

Date of Examination: July 4th 2023

Declaration of Authorship

I hereby declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified. Parts of this work have been published in:

Fabian Berressem and Arash Nikoubashman, "BoltzmaNN: Predicting effective pair potentials and equations of state using neural networks", *The Journal of Chemical Physics* **154**, 124123 (2021) <https://doi.org/10.1063/5.0045441>

Fabian Berressem, Christoph Scherer, Denis Andrienko and Arash Nikoubashman, "Ultra-Coarse-Graining of Homopolymers in Inhomogeneous Systems", *Journal of Physics: Condensed Matter* **33**, 254002 (2021) <https://doi.org/10.1088/1361-648X/abf6e2>

Place, date

Signature

Kurzzusammenfassung

Das Studium der Materialien und der Weichen Materie war und ist noch immer von großer Wichtigkeit sowohl für die moderne Forschung als auch für unser alltägliches Leben. Vor allem das Aufkommen von Computern und folglich der Etablierung von Computersimulationen hat diese Forschungsgebiete sogar noch mehr an Dynamik gewinnen lassen und zu großen Fortschritten und Errungenschaften geführt. Selbst mit modernen Rechenkapazitäten und Algorithmen bestehen jedoch weiterhin signifikante Herausforderungen bei der Untersuchung großer Systeme, bei denen mikroskopische Details das makroskopische Verhalten bedingen. Während Methoden wie *ab-initio*-Simulationen gut geeignet sind, um kleine Systeme in quantenmechanischem Detail zu behandeln, vermögen es andere Methoden wie Finite-Element-Methoden, das makroskopische Verhalten im Kontinuumslimit darzustellen. Die dazwischenliegende Skalen zeigen für sich ebenfalls interessante Phänomene, dienen aber auch dem Überbrücken der anderen Regime mittels der Methode der "gezielten Vergrößerung". In den vergangenen Jahren, haben sich durch die Verbreitung und das Studium von Maschinellen Lernen neue, datengetriebene Zugänge zur statistischen Physik und Computersimulationen eröffnet. Dieses junge Feld hat schon früh zu beeindruckenden Resultaten geführt und sich damit schnell als eine neue Säule der Natur- und Ingenieurwissenschaften etablieren können.

Diese Arbeit zielt darauf ab, einen Beitrag sowohl zum Feld der gezielten Vergrößerung als auch des Maschinellen Lernens zu leisten, und ist in zwei Teile unterteilt:

Im ersten Beitrag haben wir neuronale Netze im Bereich des direkten Designs und des inversen Designs angewendet, im Speziellen auf die Aufgaben, eine Abbildung zwischen Paarpotentialen und der resultierenden Zustandsgleichung sowie zwischen der radialen Paarverteilungsfunktion und dem zugrundeliegenden effektiven Paarpotential zu approximieren. Diese Aufgaben sind besonders interessant, da es die erste Abbildung ermöglicht, bei der Suche nach Materialien mit gewünschter Ziel-Zustandsgleichung schneller Prototypen entwickeln zu können, während die zweite Abbildung geeignet ist, etablierte, iterative Vergrößerungsmethoden zu verbessern. In beiden Unteraufgaben lag der Fokus besonders auf dem Einfluss der Darstellung der jeweiligen Ein- und Ausgaben, um trotz der geringen Menge an verfügbaren Trainingsdaten eine gute Generalisierung zu erzielen.

Der zweite Beitrag ist ein Bottom-Up Vergrößerungsschema für inhomogene Systeme, in denen ganze Polymerketten in einzelnen Kugeln zusammengefasst sind. In unserer Parametrisierung wechselwirken die vergrößerten Kugeln mittels eines Paarpotentials und zusätzlich entweder mittels eines drei-Körper Stillinger-Weber-Potentials oder eines Potentials, das von der lokalen Dichte abhängt. Es zeigt sich, dass die Kombination des Paarpotentials und des Stillinger-Weber-Potentials nicht geeignet ist, um die Eigenschaft des Referenzsystems, dünne Filme zu bilden, reproduzieren zu können. Dagegen vermögen die Systeme mit lokal-Dichte-abhängigen Wechselwirkungen, diese Eigenschaft zu erhalten und zeigen dabei sogar quantitative Übereinstimmung bei der Breite der resultierenden Filme. Bei der weiteren Untersuchung stellt sich heraus, dass es keine eindeutige Korrespondenz zwischen der Verteilung der lokalen Dichte, bezüglich der unsere Modelle optimiert sind, und den Dichteprofilen zu geben scheint. Diese Nichtkorrespondenz wird für steigende Polymerisierungsgrade und folglich zunehmende Interpenetration von Polymeren stärker, was dazu führt, dass unser Ansatz bisher lediglich für kleinere Moleküle geeignet ist. In den entsprechenden Kapiteln gehen wir auch auf mögliche Lösungsstrategien für dieses Defizit ein.

Abstract

The fields of material sciences and soft matter have been and continue to be of great importance for both modern research as well as our everyday life. Especially the advent of computers and consecutively the establishment of computer simulations caused these research areas to gain even more momentum and led to great improvements and achievements. However, even with modern computing resources and algorithms, there remain significant challenges when considering systems where microscopic details are important for the macroscopic behavior. While methods like *ab-initio*-simulations are fairly capable of treating small systems with quantum-mechanical detail, other techniques like finite-element-methods are able to capture macroscopic behavior in the continuous limit. Besides exhibiting interesting phenomena on their own, the intermediate scales are dedicated to bridging these regimes via the methodology of coarse-graining. In recent years, by virtue of machine learning becoming broadly available and a thoroughly studied field, there has been another advancement, that has opened up new, data-driven approaches to statistical physics and computer simulations. This young field has soon led to impressive results, hence establishing itself rapidly as a new pillar of sciences and engineering.

This work aims at making a contribution to both the field of coarse-graining as well as machine learning and is split in two main parts:

In the first contribution we apply neural networks for forward and inverse design, specifically to the tasks of approximating mappings from pair potentials to the resulting equation of state as well as from the radial distribution function to the effective pair potential leading to it. These tasks are very interesting as the first mapping allows for rapid prototyping when searching for materials with a desired equation of state, while the second can be used to improve established, iterative coarse-graining techniques. In both tasks, we focused mainly on the impact of the representation of the respective inputs and outputs, in order to yield good generalization capabilities despite the small number of available training examples.

The second contribution is a bottom-up coarse-graining scheme for inhomogeneous systems where whole polymer chains are mapped to single beads. In our parametrization scheme, the coarse-grained beads interact via a pair potential as well as either a three-body Stillinger-Weber potential or a local

density-dependent potential. We find that the combination of pair potential and three-body potential fails to reproduce the film-forming properties of our reference system. The systems interacting via local density-dependent potentials on the other hand are able to do so and even show quantitative agreement with regards to the width of the films. On further investigation, we find that there seems to be no unique correspondence between the distribution of the local density, which was optimized in our work, and the density profiles. This non-correspondence becomes stronger for increasing degrees of polymerization and hence increasing interpenetration of the polymer chains, which is why our approach is for now only applicable to smaller molecules. In the corresponding chapters we also elaborate on possible mitigation strategies for this shortcoming.

Contents

| | |
|--|------------|
| Declaration of Authorship | i |
| Kurzzusammenfassung | iii |
| Abstract | v |
| Table of Contents | ix |
| 1 Introduction | 1 |
| 2 Theoretical Foundations | 5 |
| 2.1 Statistical Physics and Thermodynamics | 5 |
| 2.1.1 Ensembles and Conjugated Pairs of Variables | 5 |
| 2.1.2 n -Particle Densities and n -Particle Distribution Functions | 8 |
| 2.1.3 Equations of State | 10 |
| 2.1.4 Ergodic Hypothesis | 12 |
| 2.1.5 Phase Transitions and Critical Points | 12 |
| 2.2 Polymer Physics | 13 |
| 2.2.1 Polymer Models | 16 |
| 2.2.2 Structural Properties of Polymers | 18 |
| 2.2.2.1 End-To-End Distance | 18 |
| 2.2.2.2 Kuhn Segmentation | 23 |
| 2.2.2.3 Radius of Gyration | 26 |
| 2.2.2.4 Gyration Tensor | 26 |
| 2.2.3 Scaling Behavior and Theta Temperature | 27 |
| 2.2.3.1 Monomer Interactions and Excluded Volume | 28 |
| 2.2.3.2 Solvent Qualities and Theta Temperature | 29 |
| 2.2.3.3 Scaling Behavior and Temperature Dependence | 31 |
| 2.3 Computer Simulations | 33 |
| 2.3.1 Monte Carlo Simulations | 35 |
| 2.3.2 Molecular Dynamics Simulations | 37 |
| 2.3.2.1 Integrators in MD Simulations | 38 |
| 2.3.2.2 Thermostats in MD Simulations | 40 |
| 2.3.3 Periodic Boundary Conditions | 41 |
| 2.3.4 Local Density-Dependent Potentials | 43 |

| | | |
|----------|--|-----------|
| 2.3.5 | Reduced Units | 45 |
| 2.4 | Coarse-Graining | 47 |
| 2.4.1 | Iterative Boltzmann Inversion | 48 |
| 2.4.2 | Force Matching | 49 |
| 2.4.3 | Relative Entropy Minimization | 50 |
| 2.5 | Machine Learning and Neural Networks | 52 |
| 2.5.1 | Machine Learning | 52 |
| 2.5.1.1 | Categories of Machine Learning | 52 |
| 2.5.1.2 | Training, Validation and Testing | 56 |
| 2.5.2 | Neural Networks | 56 |
| 2.5.2.1 | Artificial Neurons and Fully-Connected Layers | 57 |
| 2.5.2.2 | Convolutional Layers | 60 |
| 2.5.2.3 | Backpropagation in Neural Networks | 62 |
| 2.5.3 | Bias-Variance Tradeoff and Regularization | 64 |
| 3 | BoltzmaNN | 67 |
| 3.1 | Introduction | 68 |
| 3.2 | Methodology | 70 |
| 3.2.1 | Design of Neural Networks | 70 |
| 3.2.2 | Generation of data | 72 |
| 3.3 | Results | 73 |
| 3.3.1 | Predicting the equation of state | 73 |
| 3.3.2 | Potential prediction from pair distribution function | 79 |
| 3.3.3 | Use case examples | 88 |
| 3.4 | Conclusions | 91 |
| 4 | Ultra-Coarse-Graining | 95 |
| 4.1 | Introduction | 96 |
| 4.2 | Methods | 98 |
| 4.2.1 | Microscopically resolved reference systems | 98 |
| 4.2.2 | Coarse-graining procedure | 102 |
| 4.2.2.1 | Two-body interaction | 103 |
| 4.2.2.2 | Three-body interaction | 104 |
| 4.2.2.3 | Local density-dependent interaction | 105 |
| 4.3 | Results | 108 |
| 4.3.1 | Two-body interactions | 108 |
| 4.3.2 | Three-body SW interactions | 109 |
| 4.3.3 | Local density-dependent potentials | 110 |
| 4.3.4 | Stability of Simulations | 114 |
| 4.4 | Conclusions | 117 |

| | | |
|----------|---|------------|
| 5 | Limitiations of UCG Model | 121 |
| 5.1 | Reference Simulations and Coarse-Graining Procedure | 121 |
| 5.1.1 | Microscopically Resolved Reference Systems | 121 |
| 5.1.2 | Coarse-Graining Procedure | 124 |
| 5.2 | Analytical Expressions for Reference Properties | 126 |
| 5.3 | Analysis of Transferability and Limitations of Coarse-Grained Model | 128 |
| 5.4 | Conclusions and Outlook | 134 |
| 6 | Conclusion and Outlook | 135 |
| A | BoltzmaNN: Supplementary Information | 137 |
| A.1 | Filtering of data | 137 |
| A.2 | Virial Expansion | 137 |
| A.3 | Potential prediction from pair distribution function | 138 |
| A.4 | Schematics of network architectures | 142 |
| B | Ultra-Coarse-Graining: Supplementary Information | 145 |
| C | Glossary | 161 |
| D | Revisions and Improvements | 173 |
| | Bibliography | 189 |
| | Acknowledgments | 191 |
| | Curriculum Vitae | 193 |

CHAPTER 1 --- Introduction

The evolution of the human species has been heavily linked to the development of materials since the earliest days of humankind. This can be seen from the fact, that even some eras are named after the material that had the most prominent impact on the progress of society, *i.e.* the Stone Age, the Copper Age, the Bronze Age and the Iron Age. Besides these early contributions from metallurgy as well as other crafts working with materials like wood, bones and ceramics, countless other more recent advancements with regards to materials further shaped and improved our lives. The field of soft matter in particular has received a lot of attention in the recent centuries, due to its various contributions to agriculture, cosmetics, fuels, plastics and electronics but also fundamental research, to name just a few examples.

One milestone that heralded the age of soft matter research was the discovery of the irregular, sudden movements of particles in fluids and gases made by Scottish botanist Robert Brown in 1827, known as *Brownian motion*.¹ This observation, that had in part been made even earlier dating back even to Lucretius around 60 BC,² laid the foundation for a molecular theory of heat. Other advancements in the field of thermodynamics, mainly in the 19th and early 20th century, done by great researchers like James Clerk Maxwell, Josiah Willard Gibbs, Max Planck and Ludwig Boltzmann, led to the far-reaching successes of early thermodynamics and statistical physics, which extend beyond the field of thermodynamics to other fields like quantum mechanics, particle physics, cosmology and machine learning.

While past research on soft matter systems has led to numerous successes from theoretical and experimental physics already, it has gained even more momentum with the advent of powerful computers and efficient algorithms. The field of computer simulations that has become feasible due to this development, offers not only new insights into the behavior of the matter surrounding us but also allows for studies of unique systems, which cannot be solved analytically and where experiments still lack accuracy on the considered time and length scales. By comparing the results from simulations of computer models to the predictions from theory or experiments, it becomes possible to identify fundamental mechanisms governing certain behaviors. This in turn facilitates the development of materials exhibiting a desired behavior with a more structured

approach than plain trial and error. Hence, the field of computer simulations can be considered a third paradigm in research, complementing both theory as well as experiments. Typical approaches in this field are the so-called *forward design* where one wants to predict the properties of a given system via models and *inverse design* where one wants to find a model reproducing given quantities, both of which will be introduced in more depth in Chapter 2. Both of these applications heavily rely on the methods developed in the field of *coarse-graining*, *i.e.* the study of the systematic reduction of degrees of freedom. One of the main goals of coarse-graining is to reduce the computational complexity of a simulation in order to allow for larger systems as well as longer time scales while simultaneously reproducing qualitatively and quantitatively the actual behavior of the underlying reference system. This thesis makes a contribution in this regard by presenting results on ultra-coarse-graining of inhomogeneous systems of homopolymers using local density-dependent potentials. Furthermore, in this thesis we present work on both forward and inverse design of systems interacting via pair potentials using neural networks.

One field of research in soft matter that has been especially fascinating and successful is the study of polymers. What makes polymers so interesting to study is not only the various applications in which polymers are used but also the high intrinsic repetitiveness of their constituents, rendering them perfect systems to study using statistical methods. Due to their repetitive nature, they typically show approximately universal behaviors, which allows for studying significantly simpler, coarse-grained systems and making predictions for large systems of more realistic polymers. Besides that, polymers are ubiquitous in our world, with them being present in biological systems, like lipid bilayers and DNA, and chemical systems like plastics, so that humanity has started using polymers ever since harvesting and processing rubber from trees around 1600 BC in Mesoamerica. Since then, we have begun studying polymers more methodically in the 19th century, with early pioneering work by Henri Braconnot³ and Christian Schönbein⁴ in the 1830s when they created celluloid and cellulose acetate, which are derivatives of the natural polymer cellulose. Even though scientist have thus been working on polymer research for almost 200 years, there are still many aspects not completely understood and there is still plenty of room for development of new polymeric systems for the ever-increasing demand of innovations, which is why this field is still very active and continuously making novel contributions.

In the last century, besides the field of computer simulations there has been another addition to research paradigms, caused by the advent of *machine learning* methods. In its core, machine learning is the study of finding mappings

between given inputs and outputs by fitting typically high dimensional functions using large amounts of data. It is therefore heavily linked to data science and statistics with a lot of connections to the modeling aspects of statistical physics. The aforementioned fitting of the desired mapping can be done in a variety of different ways which is why there is a plethora of different approaches, models and problem definitions, with each having their own advantages and disadvantages for solving the countless problems machine learning has been applied to. So, instead of trying to find an analytical approximation of the desired mapping, in machine learning one is often more interested in obtaining a heuristic model derived from fitting large data sets, frequently outperforming established, classical approaches. However, this does not keep the study of machine learning from both making use of as well as contributing to a more fundamental understanding of the subject at hand. Amongst many others, one example for this can be found in this thesis, where we study not only the effect of different hyperparameters but most importantly, how transformations of the input and the target output motivated by our domain knowledge impact the performance of the employed neural networks in the specific tasks.

This rest of this thesis is structured as follows:

In Chapter 2 the theoretical foundation of this work is introduced. This chapter covers the fundamentals of statistical physics as well as computer simulations and machine learning used in the chapters going forward and begins with an overview of statistical physics and thermodynamics, followed by a short description of polymer physics. Afterwards, there is an introduction to the methodologies of computer simulations and coarse-graining. Finally, there is a short discussion of machine learning with a focus on neural networks.

In Chapter 3 we develop a machine learning method working on the problem of coarse-graining, with both an application to forward design as well as inverse design. We show that neural networks are capable of learning good approximations to equations of states for a variety of pair potentials as well as learning mappings from the radial distribution function, *i.e.* structural information, to effective pair potentials. These networks can be combined in order to predict the equation of state from a single measurement of the radial distribution function allowing for a very fast determination of macroscopic properties of a given system.

In Chapter 4 we investigate the application of local density-dependent potentials to the ultra-coarse-graining of homopolymers into single beads. While

the parametrization of inhomogeneous systems consisting of homopolymers via pair potentials exclusively is insufficient for reproducing the film-forming behavior, we find that including higher order correlations by supplementing the pair potential with an additional local density-dependent potential not only leads to stable films but is transferable for different systems as well.

In Chapter 5 we analyze extensions and limitations of the previously derived model when applied to real-world systems. We observe that for larger polymers, the model is not able to reproduce the target quantity of the reference system. We attribute this to the colloidal spheres, into which the polymers are effectively coarse-grained, becoming too soft leading to more overlap, which worsens the resolution of the structural information necessary for our coarse-graining procedure.

CHAPTER 2 --- Theoretical Foundations

The following chapter contains a discussion of the theoretical foundation necessary to understand the work described in Chapters 3 to 5. First, in Sec. 2.1 there is a short introduction to statistical physics and thermodynamics, followed by a description of polymer physics and its basic properties and results in Sec. 2.2. Then, we discuss the methodology of computer simulations in Sec. 2.3 and establish important terms and concepts of coarse-graining in Sec. 2.4. Finally, Sec. 2.5 gives an overview of machine learning with a particular focus on neural networks.

2.1 Statistical Physics and Thermodynamics

Statistical physics is a branch of physics, which aims at describing systems with very large numbers of particles using methods from statistics and probability theory. Due to the large number of particles, it is typically not possible to describe the system's evolution of *microstates*, *i.e.* the evolution of the microscopic configuration $\mathcal{C} = \{\mathbf{p}, \mathbf{q}\}$ containing all the particles' positions \mathbf{p} and momenta \mathbf{q} . However, the large number of particles allows to define *macrostates*, which are given by the macroscopic properties of the system, namely its temperature T , energy E , entropy S , pressure P , volume V , chemical potential μ and number of particles \mathcal{N} .

2.1.1 Ensembles and Conjugated Pairs of Variables

These macroscopic quantities are divided into *intensive* and *extensive* variables.⁵ For the definition of those terms, consider a system S characterized by variable X , which is divided into subsystems S_1 and S_2 characterized by X_1 and X_2 respectively. If $X = X_1 = X_2$ then X is an intensive quantity, while in the case of $X = X_1 + X_2$ the variable X is extensive. The macroscopic quantities mentioned above come in *conjugated pairs* of intensive and extensive variables, as shown in Table 2.1.

For pairs (x_j, ω_j) which are conjugated with regards to function F with the

| Intensive variable | Extensive variable |
|--------------------|--------------------|
| T | S |
| P | V |
| μ | \mathcal{N} |

Table 2.1: Conjugated pairs of intensive and extensive variables.

exact differential dF , *i.e.* a differential for which all closed path integrals vanish,

$$dF = \sum_j \omega_j dx_j, \quad \omega_j = \frac{\partial F}{\partial x_j} \quad (2.1.1.)$$

holds true. The thermodynamic variables being conjugated in combination with the *fundamental thermodynamic relation*

$$dE(S, V, \mathcal{N}) = TdS - PdV + \mu d\mathcal{N} \quad (2.1.1.2)$$

allows one to change the ensemble of independent state variables describing the system by applying Legendre transformations, which corresponds to transforming between the different thermodynamic potentials. This is useful and necessary as in thermodynamics one is typically interested in open systems, *viz.* systems which exchange heat, mass or work with their surroundings, in contrast to classical mechanics and quantum mechanics, where one typically considers isolated systems.

In this thesis, however, we consider systems in the *canonical* or \mathcal{NVT} ensemble, *i.e.* systems in thermal equilibrium with a heat bath at fixed temperature without the transfer of mass or work. For the canonical ensemble with interactions specified by the Hamiltonian $\mathcal{H}(\cdot)$, the governing thermodynamic potential is given by the free energy

$$F = E - ST = -PV + \mu\mathcal{N} \quad (2.1.1.3)$$

and the probability of a configuration $\mathcal{C} = \{\mathbf{p}, \mathbf{q}\}$ is given by

$$\mathcal{P}(\mathcal{C}) = \frac{1}{\mathcal{Z}(\beta)} e^{-\beta\mathcal{H}(\mathcal{C})} \quad (2.1.1.4)$$

with $\beta = k_B T$, Boltzmann constant k_B , and the partition function

$$\mathcal{Z}(\beta) = \int e^{-\beta\mathcal{H}(\mathcal{C})} d\mathcal{C} \quad (2.1.1.5)$$

$$= \int e^{-\beta\mathcal{H}(\{\mathbf{p},\mathbf{q}\})} d\mathbf{p}d\mathbf{q}. \quad (2.1.1.6)$$

Furthermore, it follows from the canonical probabilities in Eq. (2.1.1.4) that the expectation value of an observable $A(\cdot)$ in the canonical ensemble is given by

$$\langle A \rangle = \int \mathcal{P}(\mathcal{C}) A(\mathcal{C}) d\mathcal{C} \quad (2.1.1.7)$$

$$= \frac{1}{\mathcal{Z}(\beta)} \int e^{-\beta\mathcal{H}(\mathcal{C})} A(\mathcal{C}) d\mathcal{C}. \quad (2.1.1.8)$$

For a denumerable number of configurations, for example in the case of a discretized system with particles being positioned on lattice sites, the integration in Eqs. (2.1.1.6) and (2.1.1.8) have to be replaced by sums, yielding

$$\langle A \rangle = \sum_{\mathcal{C}} \mathcal{P}(\mathcal{C}) A(\mathcal{C}) \quad (2.1.1.9)$$

$$= \frac{\sum_{\mathcal{C}} e^{-\beta\mathcal{H}(\mathcal{C})} A(\mathcal{C})}{\sum_{\mathcal{C}} e^{-\beta\mathcal{H}(\mathcal{C})}}. \quad (2.1.1.10)$$

When assuming that the Hamiltonian can be split into the kinetic energy $K_{\mathcal{N}}(\mathbf{q})$, the interatomic potential energy $V_{\mathcal{N}}(\mathbf{p})$ and the external potential $\Phi_{\mathcal{N}}(\mathbf{p})$, *i.e.*

$$\mathcal{H}(\{\mathbf{p}, \mathbf{q}\}) = K_{\mathcal{N}}(\mathbf{q}) + V_{\mathcal{N}}(\mathbf{p}) + \Phi_{\mathcal{N}}(\mathbf{p}) \quad (2.1.1.11)$$

one can define the *configurational integral* $Z_{\mathcal{N}}$, which is given by the integral of the Boltzmann factor of the interatomic interactions over all positions,

$$Z_{\mathcal{N}}(\beta) = \int e^{-\beta V_{\mathcal{N}}(\mathbf{p})} d\mathbf{p}. \quad (2.1.1.12)$$

The assumption expressed in Eq. (2.1.1.11) is a very common and useful one, but it is not always fulfilled, *e.g.* in the case of electrically charged particles in a spatially varying external magnetic field, where $\Phi_{\mathcal{N}}$ becomes a function of the particles velocities. In the following we will use the assumption from Eq. (2.1.1.11) if not stated otherwise explicitly.

2.1.2 n -Particle Densities and n -Particle Distribution Functions

By calculating so-called n -particle densities one can obtain insights into the microscopic structure of the system at hand with varying degree of complexity. In general, n -particle densities express the probabilities of finding sets of particles like triplets, quadruplets, *etc.* at given positions $\mathbf{r} = \{\mathbf{r}^1, \mathbf{r}^2, \dots, \mathbf{r}^n\}$ and are defined as

$$\rho_{\mathcal{N}}^{(n)}(\mathbf{r}) = \left\langle \sum_{i_1=1}^{\mathcal{N}} \sum_{i_2 \neq i_1}^{\mathcal{N}} \cdots \sum_{i_n \notin \{i_1, i_2, \dots, i_{n-1}\}}^{\mathcal{N}} \prod_{j=1}^n \delta(\mathbf{r}^j - \mathbf{p}^{i_j}) \right\rangle \quad (2.1.2.1)$$

$$= \frac{\mathcal{N}!}{(\mathcal{N} - n)!} \frac{1}{Z_{\mathcal{N}}(\beta)} \int \sum_{i_1=1}^{\mathcal{N}} \sum_{i_2 \neq i_1}^{\mathcal{N}} \cdots \sum_{i_n \notin \{i_1, i_2, \dots, i_{n-1}\}}^{\mathcal{N}} \prod_{j=1}^n \delta(\mathbf{r}^j - \mathbf{p}^{i_j}) e^{-\beta V_{\mathcal{N}}(\mathbf{p})} d\mathbf{p} \quad (2.1.2.2)$$

$$= \frac{\mathcal{N}!}{(\mathcal{N} - n)!} \frac{1}{Z_{\mathcal{N}}(\beta)} \int e^{-\beta V_{\mathcal{N}}(\{\mathbf{r}^1, \dots, \mathbf{r}^n, \mathbf{p}^{n+1}, \mathbf{p}^{\mathcal{N}}\})} d\mathbf{p}^{n+1} \dots d\mathbf{p}^{\mathcal{N}} \quad (2.1.2.3)$$

where the combinatorial prefactor of $\frac{\mathcal{N}!}{(\mathcal{N}-n)!}$ comes from the interchangeability of the particles. From Eq. (2.1.2.1) it becomes clear that $\rho_{\mathcal{N}}^{(n)}(\mathbf{r})d\mathbf{r}$ yields the probability of finding n particles in the volume element $d\mathbf{r}$.

For the case of two particles, as is often used due to its simplicity, Eq. (2.1.2.1) reduces to

$$\rho_{\mathcal{N}}^{(2)}(\{\mathbf{r}^1, \mathbf{r}^2\}) = \left\langle \sum_{i=1}^{\mathcal{N}} \sum_{j \neq i}^{\mathcal{N}} \delta(\mathbf{r}^1 - \mathbf{p}^i) \delta(\mathbf{r}^2 - \mathbf{p}^j) \right\rangle \quad (2.1.2.4)$$

while the 1-particle density on the other hand of course reads

$$\rho_{\mathcal{N}}^{(1)}(\{\mathbf{r}^1\}) = \left\langle \sum_{i=1}^{\mathcal{N}} \delta(\mathbf{r}^1 - \mathbf{p}^i) \right\rangle \quad (2.1.2.5)$$

and hence describes the expected number of particles located at \mathbf{r}^1 .

Using Eqs. (2.1.2.1) and (2.1.2.5) one can define the so-called n -particle distribution functions

$$g_{\mathcal{N}}^n(\{\mathbf{r}^1, \mathbf{r}^2, \dots, \mathbf{r}^n\}) = \frac{\rho_{\mathcal{N}}^{(n)}(\{\mathbf{r}^1, \mathbf{r}^2, \dots, \mathbf{r}^n\})}{\prod_{i=1}^n \rho_{\mathcal{N}}^{(1)}(\{\mathbf{r}^i\})}. \quad (2.1.2.6)$$

These n -particle distribution functions effectively describe how correlated the particle positions are and hence, how much the structure deviates from complete randomness, *i.e.* the structure of an ideal gas. For a homogeneous system, the 1-particle density function is equal to the global density $\rho_{\mathcal{N}}^{(1)}(\mathbf{r}) = \rho = \frac{\mathcal{N}}{V}$ and the n -particle distribution functions reduce to

$$g_{\mathcal{N}}^n(\{\mathbf{r}^1, \mathbf{r}^2, \dots, \mathbf{r}^n\}) = \rho^{(-n)} \rho_{\mathcal{N}}^{(n)}(\{\mathbf{r}^1, \mathbf{r}^2, \dots, \mathbf{r}^n\}). \quad (2.1.2.7)$$

While the n -particle densities as well as the n -particle distribution functions provide a complete description of the structure of a system, it is often sufficient to know the low-order particle densities or distribution functions in order to calculate good approximations for thermodynamic properties. Furthermore, these functions can be used to describe the number of particle pairs displaced by a vector \mathbf{d} as follows:

$$\left\langle \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \sum_{j \neq i}^{\mathcal{N}} \delta(\mathbf{d} - (\mathbf{r}^i - \mathbf{r}^j)) \right\rangle = \left\langle \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \sum_{j \neq i}^{\mathcal{N}} \delta(\mathbf{d} - \mathbf{r}^i + \mathbf{r}^j) \right\rangle \quad (2.1.2.8)$$

$$= \left\langle \frac{1}{\mathcal{N}} \int \sum_{i=1}^{\mathcal{N}} \sum_{j \neq i}^{\mathcal{N}} \delta(\mathbf{d} + \mathbf{r}' - \mathbf{r}^i) \delta(\mathbf{r}' - \mathbf{r}^j) d\mathbf{r}' \right\rangle \quad (2.1.2.9)$$

$$= \frac{1}{\mathcal{N}} \int \rho_{\mathcal{N}}^{(2)}(\mathbf{r}' + \mathbf{d}, \mathbf{r}') d\mathbf{r}'. \quad (2.1.2.10)$$

For an isotropic and homogeneous system, the distribution function becomes a function of the absolute value of the displacement $d = |\mathbf{d}|$. Besides that, Eq. (2.1.2.7) can be used to simplify the above expression even further, yielding

$$\left\langle \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \sum_{j \neq i}^{\mathcal{N}} \delta(\mathbf{d} - (\mathbf{r}^i - \mathbf{r}^j)) \right\rangle = \frac{\rho^2}{\mathcal{N}} \int g_{\mathcal{N}}^2(\mathbf{d}, \mathbf{r}') d\mathbf{r}' \quad (2.1.2.11)$$

$$= \frac{\rho}{V} \int g(d) d\mathbf{r}' \quad (2.1.2.12)$$

$$= \rho g(d) \quad (2.1.2.13)$$

such that

$$g(d) = \frac{V}{\mathcal{N}^2} \left\langle \sum_{i=1}^{\mathcal{N}} \sum_{j \neq i}^{\mathcal{N}} \delta(d - |\mathbf{r}^i - \mathbf{r}^j|) \right\rangle \quad (2.1.2.14)$$

where $g(d)$ is the *radial distribution function*, also known as “RDF” or “ $g(r)$ ”. From an expansion of the n -particle distribution function of a homogeneous system as a power series of the *activity*, one can derive the *low-density approximation* of the RDF, which reads

$$\lim_{\rho \rightarrow 0} g(r) = e^{-\beta U(r)} \quad (2.1.2.15)$$

where $U(r)$ denotes the pair interaction between particles at distance r .

The radial distribution function can be obtained experimentally via scattering experiments. In these experiments, the so-called *structure factor* can be measured, which is closely related to the radial distribution function via a Fourier transformation. For a homogeneous and isotropic system the structure factor does not depend on the scattering vector \mathbf{k} but on its absolute magnitude $k = |\mathbf{k}|$ and can be expressed as

$$S(k) = 1 + \frac{1}{\mathcal{N}} \left\langle \sum_{i \neq j} e^{-i\mathbf{k}(\mathbf{r}^i - \mathbf{r}^j)} \right\rangle \quad (2.1.2.16)$$

$$= 1 + \rho \int e^{-i\mathbf{k}\mathbf{r}} g(|\mathbf{r}|) \, d\mathbf{r}. \quad (2.1.2.17)$$

2.1.3 Equations of State

An equation relating the aforementioned state variables in a given system is called the system’s *equation of state*. Typically but not necessarily these equations express the pressure as a function G of the other macroscopic quantities, *i.e.*

$$P = G(T, V, \mathcal{N}). \quad (2.1.3.1)$$

A special form of equation of state is given by the virial expansion, which expresses the pressure as a power series of the density:⁶

$$\beta P = \rho + \sum_{i=2}^{\infty} B_i(T) \rho^i \quad (2.1.3.2)$$

where $\rho = \mathcal{N}/V$ is the density and the temperature dependent coefficients B_i denote the so-called *virial coefficients*, which are determined by the interactions in the system. For homogeneous systems with isotropic pairwise interactions, the first two virial coefficients are given by

$$B_2(T) = -\frac{1}{2} \int f(|\mathbf{r}|, T) \, d\mathbf{r} \quad (2.1.3.3)$$

$$B_3(T) = -\frac{1}{3} \int \int f(|\mathbf{r}|, T) f(|\mathbf{r}'|, T) f(|\mathbf{r} - \mathbf{r}'|, T) d\mathbf{r} d\mathbf{r}' \quad (2.1.3.4)$$

where we use the Mayer- f -functions defined via

$$f(r, T) = e^{-\beta U(r)} - 1 \quad (2.1.3.5)$$

with the pair potential $U(r)$ and β as before. Higher order virial coefficients can for example be obtained via diagrammatic methods and become increasingly more complicated as they incorporate more and more correlations via combinations of Mayer- f -functions. Note that for non-interacting systems, *i.e.* $U(r) = 0$, $f(r, T)$ vanishes, $f(r, T) = 0$, and therefore all the virial coefficients vanish as well, $B_i(T) = 0 \forall i$, such that Eq. (2.1.3.2) becomes

$$\beta P = \rho = \frac{\mathcal{N}}{V} \quad (2.1.3.6)$$

which is the ideal gas equation. Hence, the virial coefficients can be seen as a measure of the deviation from ideal gas behavior.

For isotropic potentials and homogeneous systems, as assumed above, the second virial coefficient can easily be calculated from an integral in spherical coordinates:

$$B_2 = -\frac{1}{2} \int f(|\mathbf{r}|, T) d\mathbf{r} \quad (2.1.3.7)$$

$$= -\frac{1}{2} \int_0^\infty \int_0^2 \pi \int_0^\pi r^2 \sin(\theta) f(r, T) d\theta d\phi dr \quad (2.1.3.8)$$

$$= -2\pi \int_0^\infty r^2 f(r, T) dr \quad (2.1.3.9)$$

In the case of infinite dilution, so for $\rho \rightarrow 0$, higher orders of the density approximately vanish, which is why one can consider only the second virial coefficient, yielding

$$\beta P \approx \rho - 2\pi\rho^2 \int_0^\infty r^2 f(r, T) dr. \quad (2.1.3.10)$$

Using the low density approximation of the radial distribution function $g(r)$, Eq. (2.1.2.15) one can rewrite Eq. 2.1.3.10 as

$$\beta P \approx \rho - 2\pi\rho^2 \int_0^\infty r^2 (g(r) - 1) dr. \quad (2.1.3.11)$$

This means, that in the low density limit, it is sufficient to determine the structural composition or the interaction Hamiltonian of such a system in order to approximate its equation of state.

2.1.4 Ergodic Hypothesis

A very important and fundamental assumption in statistical physics is the *ergodic hypothesis*, which states that for sufficiently long times, the time average \overline{A} of a measured quantity A equals its ensemble expectation value $\langle A \rangle$,

$$\overline{A} = \lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t A(\mathcal{C}(t')) dt' = \int \mathcal{P}(\mathcal{C}) A(\mathcal{C}) d\mathcal{C} = \langle A \rangle, \quad (2.1.4.1)$$

where $\mathcal{C}(t)$ denotes the microstate at time t . The underlying assumption is that every state can be reached from every other state in a finite number of steps. This means that the phase space, that is the space of all possible microscopic configurations, is connected, *i.e.* there are no disjoint part of the phase space so that there is at least one state which is not reachable from another state. This is equivalent to the statement that for finite times the trajectory of a dynamical system comes arbitrarily close to any point in the phase space.

For denumerable many configurations, the integration on the right hand side of Eq. (2.1.4.1) has to be replaced by a sum over all possible configurations, *cf.* the discussion in Sec. 2.1.1 and Eq. (2.1.1.9) in particular. This observation is crucial especially for the field of computer simulations, as it implies that molecular dynamics and Monte-Carlo simulations, both of which will be discussed in more details in Sec. 2.3, yield the same results for the expectation values of observables. On occurrence of spontaneous symmetry breaking, for example in the case of phase transitions the ergodic hypothesis can be violated, which is called *ergodicity breaking*. In these cases, the phase space becomes disjointed such that the trajectory of the system will only cover part of the whole phase space in finite times.

2.1.5 Phase Transitions and Critical Points

In physics a *phase* corresponds to a region of a system in which all physical properties are essentially equal. Besides the common distinction into solid, liquid and gas phases, physical phases are not limited to just these three categories. For instance, a system in a solid state might even exhibit multiple different structures under different conditions, which is the case for example for the 19 different known solid crystalline phases of water. Hence, there can be numerous different phases for systems consisting of only one type of molecule but the number of possible phases typically further increases for mixtures of components, *e.g.* hydrogels, *i.e.* mixtures of water and hydrophilic polymers which do not dissolve in water.

A *phase transition* occurs, when the systems transitions from one phase to another, where the phases are distinguished by an *order parameter*. This order parameter is a function describing the order, where in one phase the order parameter typically equals zero and in the other phase it is non-zero. An example for such an order parameter is the magnetization of a ferromagnetic system which is subject to an external field: below the *critical temperature* T_c the system has a net magnetization, while above this temperature the magnetization is zero. In this example, the temperature functions as the *control parameter* of this transition, *i.e.* the temperature is the parameter that is varied such that the system transitions between the two phases.

Besides that, the system can not only transition between phases, but exhibit multiple phases at the same time, which is called *phase coexistence*. At phase coexistence, multiple phases occur in the system simultaneously with the molecules in the system being able to cross the phase boundaries freely, *i.e.* not inhibited by any external constraint like a wall. In this case, the order parameter has to be a local quantity in order for it to distinguish the phases. A typical case for phase coexistence is a closed system in which water is boiled, which leads to the coexistence of a fluid phase and a gas phase.

By plotting the different phases as a function of the thermodynamic variables, one obtains a *phase diagram*. Hence, a phase diagram is a visualization of the phase behavior of a system given the thermodynamic conditions, with the lines drawn in these diagrams denoting the conditions for the emergence of phase coexistence. In Fig. 2.1 there is a schematic example of a typical phase diagram for a monoatomic system interacting via the Lennard-Jones potential, *cf.* Eq. (2.2.1.3), in the $\rho - T$ -plane.

In this diagram, both the *triple point* as well as the *critical point* are marked with red symbols. The triple point describes the conditions under which there is a phase coexistence of three phases, solid, liquid and gas. The critical point denotes the point in the phase diagram above which the gas and liquid phase are not distinct anymore as long as the pressure is not high enough to compact it to a solid. This *supercritical* phase shows characteristics of both gas, like a viscosity typical for gases, and liquids, like a density typical for liquids.

2.2 Polymer Physics

Polymers are macromolecules consisting of a large number of chemically bonded, identical repeating subunits, so-called *monomers*. As the term “monomer” may refer to any repeat unit along the chain, it is ambiguous by definition and one

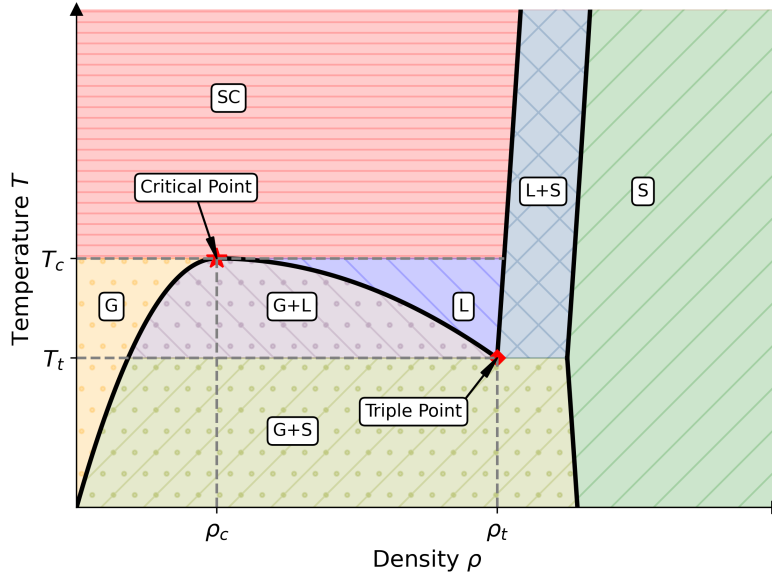


Figure 2.1: Schematic phase diagram of a monoatomic Lennard-Jones system in the $\rho - T$ -plane, cf. Eq. (2.2.1.3). “G” denotes “gas”, “L” denotes “liquid”, “S” denotes “solid”. “SC” denotes “supercritical”. The triple point is located at (ρ_t, T_t) and the critical point at (ρ_c, T_c) .

should always state a monomer’s composition explicitly. The *chemical monomer* is given by the (typically small) repeat units that were linked together via covalent bonds to make the chain. For example, the most common plastic in use today, polyethylene, is made up of chemical monomers of (C_2H_4) , while the chemical monomer of polystyrene comprises of (C_8H_8) .

On the other hand, when modeling macromolecules as polymers, one may define the monomers differently from the chemical one, so that a monomer can in principle be arbitrarily large, with a special choice of monomer being the *Kuhn monomer* discussed in Sec. 2.2.2.2. For instance, in the case of DNA, typical choices of monomers can contain a few thousand base-pairs instead of just a few atoms, as would be the case for its chemical monomer.

The number of monomers in a single polymer, also called *degree of polymerization*, is typically very large and can even reach numbers of 10^{10} , depending on the definition of the monomers. Polymers are often treated using *bead-spring-models*, i.e. the monomers are considered as beads connected by springs, which behave as effective bonds of the monomers.

Figure 2.2 shows a simulation snapshot of polystyrene, with aromatic carbon atoms shown in red and non-aromatic ones in green. The monomeric repeat

units (C_8H_8) are indicated by grey beads located at the corresponding centers of masses and connected by grey bonds.

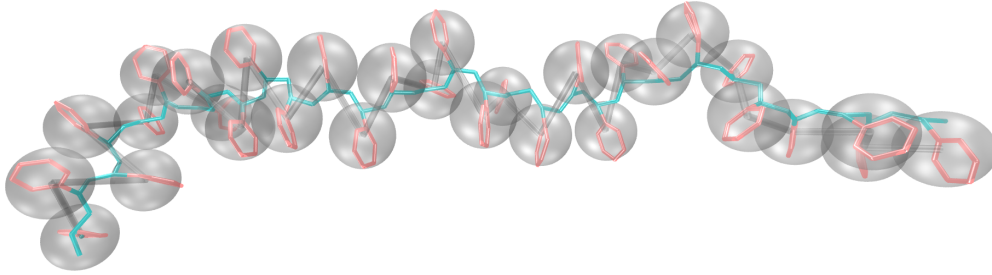


Figure 2.2: Schematic of polystyrene and its representation as a bead-spring model. The red color indicates aromatic carbon atoms and green color indicates non-aromatic ones. The grey beads represent the (C_8H_8)-monomers and are located at the corresponding centers of mass. The grey bonds connecting these beads are effective bonds and are typically treated as springs.

The large size of the polymers as well as the identical composition of the monomers typically make it feasible to treat polymers using statistical methods as they approximately exhibit a universality in their behavior for sufficiently large molecules. Polymers offer a vast chemical configuration space, as they can come in a variety of possible shapes, such as linear polymers, branched polymers and star-polymers, and can consist of different monomer types or even combinations of two or more different monomer types, forming so-called *copolymers*. The *molecular weight* of a copolymer is then given by

$$M = \sum_{\alpha \in \mathcal{T}} m_{\alpha} N_{\alpha}, \quad (2.2.0.1)$$

where \mathcal{T} denotes the set of occurring monomer types, m_{α} the mass of monomers of type α and N_{α} the number of monomers of type α . Besides the variety of possible compositions and structures of single polymers, one can investigate mixtures of multiple polymers, which makes this chemical configuration space and the number of possible applications even more intractable.

While there are many different shapes for polymers, this thesis will treat only linear chains and the introduction provided below always refers to this type of polymer, if not stated otherwise explicitly.

There exists a plethora of different models for polymers which vary in their complexity and the atomistic properties they retain. Typically, models which include interactions between the monomers are very hard to treat analytically,

which is why one often resorts to models without explicit interactions, so called *ideal polymer models*. Even though this simplification is a very strong one, ideal polymer models are oftentimes very successful in capturing the qualitative and quantitative behavior of polymers, leading to a wide range of physical insight especially for studying conformations of single chains. In the following section we will present some of the ideal models of polymers as well as the *Kremer-Grest model*,^{7,8} a simple but effective model containing explicit interactions which was used in a modified version in Chapters 4 and 5.

2.2.1 Polymer Models

The simplest polymer model is that of a *freely jointed chain*. In this model there are no interactions between monomers as well as no restrictions on the flexibility of the chain. The only constraint in this model is given by the bonds having a fixed length. Hence, the statistics follows that of a *random walk* with equal step size in all dimensions.¹ Quantities derived for this model will be denoted with a superscript “*fj*”.

The *freely rotating chain* is similar to the freely jointed chain in that it restricts the bonds to a fixed length. Furthermore, it has the additional constraint that all pairs of neighboring bonds make a fixed angle. This effectively poses a constraint on the flexibility of the chain and therefore incorporates stiffness in the polymer model. Quantities derived for this model will be denoted with a superscript “*fr*”.

The *gaussian chain* model relaxes the constraint of fixed bond lengths by modeling the components of the bond vector \mathbf{b} by normal distributions around 0, $(\mathbf{b})_k = \mathcal{N}(\mu = 0, \sigma^2 = \frac{b^2}{3})$, where, as one can calculate from the probability distribution of the length of \mathbf{b} , b is the mean bond length. This is a more realistic assumption than stiff bonds, as in real world systems bonds are not exactly stiff as well, while still allowing for tuning of the expectation value of the bond length via the standard deviation of the normal distribution.

In the *worm-like chain* model, one introduces a persistence length P , which limits the chain flexibility by adding an energy term to the Hamiltonian describing the energy needed for bending, $E^{\text{bend}}[\mathbf{R}(\cdot)] = \frac{k_B T}{2} \int_0^L P \left(\frac{\partial^2 \mathbf{R}(s)}{\partial s^2} \right) ds$, where L is the polymers length and the integral is a contour integral, with the

¹Throughout this introduction we will assume three spatial dimensions, if not stated otherwise explicitly.

polymer's contour given by $\mathbf{R}(\cdot)$.

While the ideal polymer models described above do not incorporate interactions between monomers, the Kremer-Grest model as described in Ref. [7] and [8] does include both bonded as well as non-bonded interactions. The bonded interactions are given by a FENE potential,⁹ which stands for “finitely extensible nonlinear elastic”,

$$U^{\text{FENE}}(r_{ij}) = \begin{cases} -\frac{1}{2}kr_0^2 \ln [1 - (r_{ij}/r_0)^2], & r_{ij} < r_0 \\ \infty, & r_{ij} \geq r_0 \end{cases} \quad (2.2.1.1)$$

with spring constant $k = 30 k_B T / \sigma^2$, maximum bond extension $r_0 = 1.5 \sigma$ and the distance r_{ij} between bonded particles i and j .

The non-bonded interactions, *i.e.* those that apply to all pairs of monomers, regardless of bonds, are given by a Week-Chandler-Andersen potential¹⁰, also known as WCA potential, which reads

$$U^{\text{WCA}}(r_{ij}) = \begin{cases} 4\varepsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 + \frac{1}{4} \right], & r_{ij} < 2^{1/6}\sigma \\ 0, & r_{ij} \geq 2^{1/6}\sigma. \end{cases} \quad (2.2.1.2)$$

Here ε determines the interaction strength, while σ is the size parameter, modeling the *excluded volume* of the monomers, *viz.* the volume, that is occupied by a monomer such that no other monomer can intrude this volume. The constant shift in the potential for distances smaller than $2^{1/6}\sigma$ is included to ensure the smoothness of the potential and avoid an unphysical potential jump, which would cause an infinitely large force peak at $r_{ij} = 2^{1/6}\sigma$.

The WCA potential is a special case of the Lennard-Jones potential^{11–13} or LJ potential, where the attractive part of the potential is not discarded like in the case of the WCA potential but replaced by a cutoff at r_c :

$$U^{\text{LJ}}(r_{ij}) = \begin{cases} 4\varepsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right], & r_{ij} < r_c \\ 0, & r_{ij} \geq r_c. \end{cases} \quad (2.2.1.3)$$

By inclusion of an attraction, the model is capable of capturing the film forming properties of polymers, which would not be reproduced by purely repulsive interactions.

The shift of the potential has been dismissed in Eq. (2.2.1.3) and can be replaced by other means of dealing with discontinuities at the cutoff. In this work, we used a version of the LJ potential, where the potential is simply truncated at the cutoff as well as one, where a smoothing function is applied.

2.2.2 Structural Properties of Polymers

In this section, we will define and explain some of the structural properties of polymers that are important for this work.

2.2.2.1 End-To-End Distance

Some of the structural properties of polymers can be calculated and investigated analytically for ideal chain models. One example for such a property is the *end-to-end-distance* \mathbf{R}_{ee} which is given by the vector between the first and the last monomer. So, for a polymer consisting of N monomers at positions $\mathbf{R} = \{\mathbf{r}_i | i \in \{1, \dots, N\}\}$, the end-to-end distance reads

$$\mathbf{R}_{ee}(\mathbf{R}) = \sum_{i=1}^{N-1} (\mathbf{r}_{i+1} - \mathbf{r}_i) \quad (2.2.2.1)$$

$$= \mathbf{r}_N - \mathbf{r}_1 \quad (2.2.2.2)$$

where the second equality stems from all other terms canceling out. For infinite times,² the expectation value of \mathbf{R}_{ee} is of course zero for homogeneous and isotropic systems, as the polymers orientations during the course of this time will average out. Hence, it is useful, to determine instead the *average squared end-to-end distance* $\langle \mathbf{R}_{ee}^2 \rangle$. The average squared end-to-end distance provides information about the polymers extent and is defined straight-forward from

$$\langle \mathbf{R}_{ee}^2 \rangle = \sum_{\mathbf{R} \in \mathcal{R}} \mathcal{P}(\mathbf{R}) (\mathbf{R}_{ee}(\mathbf{R}))^2 \quad (2.2.2.3)$$

with $\mathcal{R} = \{\mathbf{R}\}$ denoting the set of all polymer configurations and $\mathcal{P}(\mathbf{R})$ the corresponding probabilities of said configurations. The square of $\mathbf{R}_{ee}(\mathbf{R})$ refers to the squared absolute value of the respective vector as calculated using the standard scalar product “ \cdot ”, *i.e.*

$$(\mathbf{R}_{ee}(\mathbf{R}))^2 = \mathbf{R}_{ee}(\mathbf{R}) \cdot \mathbf{R}_{ee}(\mathbf{R}) \quad (2.2.2.4)$$

$$= |\mathbf{R}_{ee}(\mathbf{R})|^2 \quad (2.2.2.5)$$

$$= (\mathbf{R}_{ee}(\mathbf{R}))_x^2 + (\mathbf{R}_{ee}(\mathbf{R}))_y^2 + (\mathbf{R}_{ee}(\mathbf{R}))_z^2. \quad (2.2.2.6)$$

For ideal chains this quantity can easily be calculated analytically, as will be shown for the example of a freely jointed chain.

²or equivalently for an infinite number of configurations, as stated by the ergodicity theorem

For the freely jointed chain the distance between neighboring monomers i and $i + 1$ can be expressed as

$$\mathbf{r}_{i+1} - \mathbf{r}_i = \mathbf{b}_i = b\hat{\mathbf{e}}_i \quad (2.2.2.7)$$

with b denoting the constant bond length and $\hat{\mathbf{e}}_i$ being the unit vector pointing from \mathbf{r}_i to \mathbf{r}_{i+1} . As there are no constraints imposed in the freely jointed chain model besides the constant bond lengths, there are no restrictions to $\hat{\mathbf{e}}_i$ so that its orientation is randomly distributed. This implies in particular, that

$$\langle \hat{\mathbf{e}}_i \cdot \hat{\mathbf{e}}_j \rangle = \delta_{i,j}. \quad (2.2.2.8)$$

where $\delta_{i,j}$ refers to the Kronecker-delta, which equals one for $i = j$ and zero otherwise.

Hence, using Eq. (2.2.2.1) the average squared end-to-end distance for this model, indicated by a superscript “ fj ”, can be calculated as

$$\langle (\mathbf{R}_{ee}^{fj})^2 \rangle = \langle \mathbf{R}_{ee}^{fj} \cdot \mathbf{R}_{ee}^{fj} \rangle \quad (2.2.2.9)$$

$$= \left\langle \left(\sum_{i=1}^{N-1} \mathbf{b}_i \right) \cdot \left(\sum_{j=1}^{N-1} \mathbf{b}_j \right) \right\rangle \quad (2.2.2.10)$$

$$= \left\langle \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} (\mathbf{b}_i \cdot \mathbf{b}_j) \right\rangle \quad (2.2.2.11)$$

$$= \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \langle \mathbf{b}_i \cdot \mathbf{b}_j \rangle \quad (2.2.2.12)$$

$$= b^2 \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \langle \hat{\mathbf{e}}_i \cdot \hat{\mathbf{e}}_j \rangle \quad (2.2.2.13)$$

$$= b^2 \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \langle \cos(\Theta_{ij}) \rangle \quad (2.2.2.14)$$

$$= b^2 \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \delta_{i,j} \quad (2.2.2.15)$$

$$= b^2 \sum_{i=1}^{N-1} 1 \quad (2.2.2.16)$$

$$= b^2(N - 1) \quad (2.2.2.17)$$

where Θ_{ij} denotes the angle between bond vectors \mathbf{b}_i and \mathbf{b}_j .

While fairly simple in its derivation and its statement, Equation (2.2.2.17) is a fundamentally important result in that it shows, that the size of the polymer grows not with the number of monomers, but with its square root. Furthermore, this equation is used for the definition of the so-called *Kuhn length*¹⁴ b_k . In the *Kuhn segmentation*, see Sec. 2.2.2.2, the polymer is mapped to an *equivalent freely jointed chain* with the same mean squared end-to-end distance and maximum end-to-end distance, with the resulting segment length denoted by b_k . This quantity defines a relevant length scale of the system and describes how strongly correlated neighboring monomer positions are, *viz.* the length along the chain after which the bond vectors are approximately uncorrelated. This becomes more clear when considering the mean squared end-to-end distance of the freely rotating chain, denoted with a subscript “*fr*”.

For the freely rotating chain, both the bond length b and the angle θ between two adjacent bond vectors are fixed, *i.e.*

$$\mathbf{r}_{i+1} - \mathbf{r}_i = \mathbf{b}_i = b \hat{\mathbf{e}}_i \quad (2.2.2.18)$$

$$\hat{\mathbf{e}}_i \cdot \hat{\mathbf{e}}_{i+1} = \cos(\theta). \quad (2.2.2.19)$$

Furthermore, the expectation value of a bond vector can be expressed in terms of its average projection on its predecessor, which can then be done repeatedly:

$$\langle \mathbf{b}_i \rangle = \frac{\langle \mathbf{b}_i \cdot \mathbf{b}_{i-1} \rangle \mathbf{b}_{i-1}}{|\mathbf{b}_{i-1}|^2} \quad (2.2.2.20)$$

$$= \frac{\langle \mathbf{b}_i \cdot \mathbf{b}_{i-1} \rangle \langle \mathbf{b}_{i-1} \cdot \mathbf{b}_{i-2} \rangle \mathbf{b}_{i-2}}{|\mathbf{b}_{i-1}|^2 |\mathbf{b}_{i-2}|^2} \quad (2.2.2.21)$$

$$\dots \quad (2.2.2.22)$$

$$= \frac{\langle \mathbf{b}_i \cdot \mathbf{b}_{i-1} \rangle \langle \mathbf{b}_{i-1} \cdot \mathbf{b}_{i-2} \rangle \dots \langle \mathbf{b}_{i-j+1} \cdot \mathbf{b}_{i-j} \rangle \mathbf{b}_{i-j}}{|\mathbf{b}_{i-1}|^2 |\mathbf{b}_{i-2}|^2 \dots |\mathbf{b}_{i-j}|^2} \quad (2.2.2.23)$$

$$= \left(\prod_{k=1}^j \frac{\langle \mathbf{b}_{i-k} \cdot \mathbf{b}_{i-k-1} \rangle}{|\mathbf{b}_{i-k-1}|^2} \right) \mathbf{b}_{i-j} \quad (2.2.2.24)$$

As the bonds have constant length b and with the scalar product of adjacent bond orientation vectors as given by Eq. (2.2.2.19), this expression simplifies to

$$\langle \mathbf{b}_i \rangle = \left(\prod_{k=1}^j \frac{b^2 \cos(\theta)}{b^2} \right) \mathbf{b}_{i-j} \quad (2.2.2.25)$$

$$= \frac{b^{2j} \cos(\theta)^j}{b^{2j}} \mathbf{b}_{i-j} \quad (2.2.2.26)$$

$$= \cos(\theta)^j \mathbf{b}_{i-j}. \quad (2.2.2.27)$$

This reasoning also holds for the expectation value of a bond vector expressed in terms of its successor, yielding

$$\langle \mathbf{b}_i \rangle = \cos(\theta)^j \mathbf{b}_{i+j} \quad (2.2.2.28)$$

so in general,

$$\langle \mathbf{b}_i \rangle = \cos(\theta)^{|j|} \mathbf{b}_{i+j} \quad (2.2.2.29)$$

holds for all j such that $1 \leq i + j \leq N$. From Eq. (2.2.2.29) and the linearity of the average it follows that

$$\langle \mathbf{b}_i \cdot \mathbf{b}_j \rangle = \langle \cos(\theta)^{|i-j|} \mathbf{b}_j \cdot \mathbf{b}_j \rangle \quad (2.2.2.30)$$

$$= \langle \cos(\theta)^{|i-j|} b^2 \rangle \quad (2.2.2.31)$$

$$= b^2 \cos(\theta)^{|i-j|}. \quad (2.2.2.32)$$

Using Eq. (2.2.2.32) in the definition of the average squared end-to-end distance yields

$$\langle (\mathbf{R}_{ee}^{fr})^2 \rangle = \langle \mathbf{R}_{ee}^{fr} \cdot \mathbf{R}_{ee}^{fr} \rangle \quad (2.2.2.33)$$

$$= \left\langle \left(\sum_{i=1}^{N-1} \mathbf{b}_i \right) \cdot \left(\sum_{j=1}^{N-1} \mathbf{b}_j \right) \right\rangle \quad (2.2.2.34)$$

$$= \left\langle \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \mathbf{b}_i \cdot \mathbf{b}_j \right\rangle \quad (2.2.2.35)$$

$$= \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \langle \mathbf{b}_i \cdot \mathbf{b}_j \rangle \quad (2.2.2.36)$$

$$= b^2 \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \langle \cos(\Theta_{ij}) \rangle \quad (2.2.2.37)$$

$$= b^2 \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \cos(\theta)^{|i-j|} \quad (2.2.2.38)$$

$$= b^2 \sum_{i=1}^{N-1} \left(1 + \sum_{j=1}^{i-1} \cos(\theta)^{|i-j|} + \sum_{j=i+1}^{N-1} \cos(\theta)^{|i-j|} \right) \quad (2.2.2.39)$$

$$= b^2(N-1) + b^2 \sum_{i=1}^{N-1} \left(\sum_{j=1}^{i-1} \cos(\theta)^{|i-j|} + \sum_{j=i+1}^{N-1} \cos(\theta)^{|i-j|} \right) \quad (2.2.2.40)$$

with Θ_{ij} being the angle between \mathbf{b}_i and \mathbf{b}_j as before. As $|\cos(\theta)| < 1$ for all values except for multiples of π , which would correspond to a chain that continuously maps back to itself or a straight line, and using the fact that N is large, $N \gg 1$, one can use the following approximation including the geometric series:

$$\sum_{i=1}^{N-1} \left(\sum_{j=1}^{i-1} \cos(\theta)^{|i-j|} + \sum_{j=i+1}^{N-1} \cos(\theta)^{|i-j|} \right) \approx 2 \sum_{i=1}^{N-1} \sum_{j=1}^{\infty} \cos(\theta)^j \quad (2.2.2.41)$$

$$= 2 \sum_{i=1}^{N-1} \frac{\cos(\theta)}{1 - \cos(\theta)} \quad (2.2.2.42)$$

$$= 2(N-1) \frac{\cos(\theta)}{1 - \cos(\theta)}. \quad (2.2.2.43)$$

Employing this expression in Eq. (2.2.2.40) leads to

$$\langle (\mathbf{R}_{ee}^{fr})^2 \rangle = b^2(N-1) + 2b^2(N-1) \frac{\cos(\theta)}{1 - \cos(\theta)} \quad (2.2.2.44)$$

$$= b^2(N-1) \frac{1 + \cos(\theta)}{1 - \cos(\theta)}. \quad (2.2.2.45)$$

Eqs. (2.2.2.14) and (2.2.2.37) hold for all ideal chains with constant bond length. Furthermore, for these models the correlations between bond vectors \mathbf{b}_i and \mathbf{b}_j vanish if i and j are far away along the chain, due to the absence of interactions between monomers, implying

$$\lim_{|i-j| \rightarrow \infty} \langle \cos(\Theta_{ij}) \rangle = 0. \quad (2.2.2.46)$$

It can be shown that the sum over all bond vectors \mathbf{b}_j given any bond vector \mathbf{b}_i converges, so that one can define

$$C'_i = \sum_{j=1}^{N-1} \langle \cos(\Theta_{ij}) \rangle. \quad (2.2.2.47)$$

Using Eq. (2.2.2.47) one can express the mean squared end-to-end distance for ideal chains as

$$\langle (\mathbf{R}_{ee})^2 \rangle = b^2 \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \langle \cos(\Theta_{ij}) \rangle \quad (2.2.2.48)$$

$$= b^2 \sum_{i=1}^{N-1} C'_i \quad (2.2.2.49)$$

$$= C_N(N-1)b^2 \quad (2.2.2.50)$$

where

$$C_N = \frac{1}{N-1} \sum_{i=1}^{N-1} C'_i \quad (2.2.2.51)$$

is the average of all the coefficients C'_i and is called *Flory's characteristic ratio*. For long chains, $N \rightarrow \infty$, this ratio C_N converges to a finite value C_∞ so that the mean squared end-to-end distance can be approximated as

$$\langle (\mathbf{R}_{ee})^2 \rangle = C_\infty(N-1)b^2. \quad (2.2.2.52)$$

For example, comparing Eqs. (2.2.2.45) and (2.2.2.52) yields

$$C_\infty = \frac{1 + \cos(\theta)}{1 - \cos(\theta)} \quad (2.2.2.53)$$

for Flory's characteristic ratio of the freely rotating chain.

In general, Flory's characteristic ratio describes the change in a polymer's size compared to the size of a freely jointed chain with equal number of monomers and bond length due to the additional interactions or applied constraints. As the constraints might influence the enlargement or shrinking of the polymer differently, the characteristic ratio typically varies from model to model and does not need to be a constant.

2.2.2.2 Kuhn Segmentation

The segmentation of a chain into *Kuhn segments*¹⁴ is defined by mapping a polymer chain to an equivalent freely jointed chain with N_k Kuhn monomers and Kuhn length b_k such that both the maximum end-to-end distance, R_{\max} , also known as the contour length, as well as the mean squared end-to-end distance, $\langle (\mathbf{R}_{ee})^2 \rangle$, coincide. Therefore, the defining relations of the Kuhn segmentation of an ideal chain with bond length b and number of monomers N are given by

$$b_k (N_k - 1) \stackrel{!}{=} R_{\max} \quad (2.2.2.54)$$

$$b_k^2 (N_k - 1) \stackrel{!}{=} \langle \mathbf{R}_{ee}^2 \rangle \quad (2.2.2.55)$$

$$= b_k R_{\max} \quad (2.2.2.56)$$

$$= b^2 (N - 1)C_\infty \quad (2.2.2.57)$$

where the last relation holds approximately for large ideal chains, *i.e.* $N \rightarrow \infty$.

The above equations imply that the Kuhn segmentation is a form of coarse-graining in that it maps a finely resolved chain into a coarser one with multiple monomers combined into effective beads, the Kuhn monomers, such that two properties of the chain are retained, namely the maximum extension of the chains and its mean squared end-to-end distance.

From Eqs. (2.2.2.54) to (2.2.2.57), the number of Kuhn segments $N_k - 1$, *i.e.* the number of bonds in the equivalent freely jointed chain, can be calculated as

$$\frac{N_k - 1}{R_{\max}^2} = \frac{N_k - 1}{(N_k - 1)^2 b_k^2} \quad (2.2.2.58)$$

$$= \frac{1}{(N_k - 1) b_k^2} \quad (2.2.2.59)$$

$$= \frac{1}{b^2 (N - 1) C_\infty} \quad (2.2.2.60)$$

$$\Leftrightarrow N_k - 1 = \frac{R_{\max}^2}{b^2 (N - 1) C_\infty} \quad (2.2.2.61)$$

and the Kuhn length is given by

$$b_k = \frac{\langle \mathbf{R}_{ee}^2 \rangle}{R_{\max}} \quad (2.2.2.62)$$

$$= \frac{b^2 (N - 1) C_\infty}{R_{\max}}. \quad (2.2.2.63)$$

In the case of a freely jointed chain the maximum extension of the chain is given by a fully stretched chain, $R_{\max} = (N - 1)b$ and Flory's characteristic ratio is equal to unity, $C_\infty = 1$, so the mapping is trivial:

$$N_k - 1 = \frac{R_{\max}^2}{b^2 (N - 1) C_\infty} \quad (2.2.2.64)$$

$$= \frac{(N - 1)^2 b^2}{b^2 (N - 1)} \quad (2.2.2.65)$$

$$= N - 1 \quad (2.2.2.66)$$

$$b_k = \frac{b^2 (N - 1) C_\infty}{R_{\max}} \quad (2.2.2.67)$$

$$= \frac{b^2 (N - 1)}{(N - 1)b} \quad (2.2.2.68)$$

$$= b \quad (2.2.2.69)$$

Hence, in this case the coarse-graining procedure is an identity mapping, which is what one would expect.

For a freely rotation chain, the maximum extension of the chain is obtained when the bonds are arranged in a rod-like zig-zag formation, *i.e.* the bonds are all in a plane and the angles between them are alternating, which is commonly known as the *trans* state. The maximum end-to-end distance can then be calculated as $(N - 1)$ times the length of a bond projected along the contour, $b \cos\left(\frac{\theta}{2}\right)$, yielding

$$R_{\max} = (N - 1)b \cos\left(\frac{\theta}{2}\right). \quad (2.2.2.70)$$

Therefore, with Flory's characteristic ratio taken from Eq. (2.2.2.53), an equivalent freely jointed chain for this polymer model has

$$N_k - 1 = \frac{R_{\max}^2}{b^2 (N - 1)C_{\infty}} \quad (2.2.2.71)$$

$$= \frac{(N - 1)^2 b^2 \cos^2\left(\frac{\theta}{2}\right)}{b^2 (N - 1) \frac{1 + \cos(\theta)}{1 - \cos(\theta)}} \quad (2.2.2.72)$$

$$= (N - 1) \cos^2\left(\frac{\theta}{2}\right) \frac{1 - \cos(\theta)}{1 + \cos(\theta)} \quad (2.2.2.73)$$

$$= (N - 1) \frac{1}{2} (1 + \cos(\theta)) \frac{1 - \cos(\theta)}{1 + \cos(\theta)} \quad (2.2.2.74)$$

$$= (N - 1) \frac{1 - \cos(\theta)}{2} \quad (2.2.2.75)$$

Kuhn segments each with a Kuhn length of

$$b_k = \frac{b^2 (N - 1)C_{\infty}}{R_{\max}} \quad (2.2.2.76)$$

$$= \frac{b^2 (N - 1) \frac{1 + \cos(\theta)}{1 - \cos(\theta)}}{(N - 1)b \cos\left(\frac{\theta}{2}\right)} \quad (2.2.2.77)$$

$$= b \frac{1 + \cos(\theta)}{(1 - \cos(\theta)) \cos\left(\frac{\theta}{2}\right)} \quad (2.2.2.78)$$

$$= b \frac{2 \cos\left(\frac{\theta}{2}\right)}{1 - \cos(\theta)}. \quad (2.2.2.79)$$

2.2.2.3 Radius of Gyration

Besides the average squared end-to-end distance one often resorts to the *squared radius of gyration*, or R_g^2 in order to describe a polymers size. The squared radius of gyration describes the polymers extent by the variance of its monomer positions around its center of mass weighted by their masses:

$$R_g^2(\mathbf{R}) = \frac{1}{M} \sum_{i=1}^N m_i (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R}))^2 \quad (2.2.2.80)$$

where $\mathbf{R} = \{\mathbf{r}_i | i \in \{1, \dots, N\}\}$ is the set of monomer positions \mathbf{r}_i , m_i the monomers' masses and N the number of monomers. Furthermore, \mathbf{R}_{com} denotes the position of the center of mass, *i.e.*

$$\mathbf{R}_{\text{com}}(\mathbf{R}) = \frac{1}{M} \sum_{i=1}^N m_i \mathbf{r}_i \quad (2.2.2.81)$$

In the case of monomers with equal mass, in particular for polymers consisting of a single type of monomers, Eq. (2.2.2.80) simplifies to

$$R_g^2(\mathbf{R}) = \frac{1}{N} \sum_{i=1}^N (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R}))^2 \quad (2.2.2.82)$$

For ideal chains one finds that the average squared end-to-end distance and the squared radius of gyration are related via

$$R_g^2 = \frac{1}{6} \langle \mathbf{R}_{ee}^2 \rangle \quad (2.2.2.83)$$

2.2.2.4 Gyration Tensor

The squared radius of gyration can furthermore be generalized to the *gyration tensor* \mathbf{G} ,¹⁵ which is defined as

$$\mathbf{G} = \frac{1}{N} \sum_{i=1}^N \Delta \mathbf{d}_i \otimes \Delta \mathbf{d}_i \quad (2.2.2.84)$$

$$= \frac{1}{N} \sum_{i=1}^N (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R})) \otimes (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R})) \quad (2.2.2.85)$$

where $\Delta \mathbf{d}_i = \mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R})$ is the vector from the center of mass to the position of monomer i and “ \otimes ” denotes the dyadic product. Hence, for the $\alpha\beta$ component with $\alpha, \beta \in \{x, y, z\}$ of the gyration tensor, one has

$$(\mathbf{G})_{\alpha\beta} = \left(\frac{1}{N} \sum_{i=1}^N (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R})) \otimes (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R})) \right)_{\alpha\beta} \quad (2.2.2.86)$$

$$= \frac{1}{N} \sum_{i=1}^N ((\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R})) \otimes (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R})))_{\alpha\beta} \quad (2.2.2.87)$$

$$= \frac{1}{N} \sum_{i=1}^N (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R}))_{\alpha} (\mathbf{r}_i - \mathbf{R}_{\text{com}}(\mathbf{R}))_{\beta} \quad (2.2.2.88)$$

From Eqs. (2.2.2.82) and (2.2.2.88), one can calculate the radius of gyration from the gyration tensor

$$R_g = \sqrt{\mathbf{G}_{xx} + \mathbf{G}_{yy} + \mathbf{G}_{zz}}. \quad (2.2.2.89)$$

Apart from the information about the average space occupied by a chain, the gyration tensor further conveys insights about the orientation of the polymer and its deviations from a spherical shape. Cases, in which the components of the gyration tensor averaged over all configurations of a single chain fulfill $\langle \mathbf{G} \rangle_{xx} \approx \langle \mathbf{G} \rangle_{yy} \approx \langle \mathbf{G} \rangle_{zz}$, indicate that the system is isotropic, *i.e.* the chain does not have a preferred direction along which it expands. On the other hand, cases in which these components differ significantly, *e.g.* $\langle \mathbf{G} \rangle_{xx} \neq \langle \mathbf{G} \rangle_{yy} \approx \langle \mathbf{G} \rangle_{zz}$, indicate a preferred orientation of the chain along a certain axis. Furthermore, by looking at the averages of the eigenvalues of the gyration tensor, one can draw conclusions on the average shape of the polymer, with similar eigenvalues suggesting a more spherical shape and differing eigenvalues suggesting an elongation of the polymer. This is especially useful in anisotropic systems with many chains, where there is a preferred orientation of the chains, or inhomogeneous systems, where the shapes of the spheres will change depending on their position, both of which manifest themselves in the components of $\langle \mathbf{G} \rangle$ or its eigenvalues respectively.

2.2.3 Scaling Behavior and Theta Temperature

The ideal polymer models discussed above give valuable qualitative and often-times quantitative insights into the properties of polymers, yet are very easy to treat analytically. In reality however, the interactions between monomers are

not negligible so that real chains exhibit far more complex behaviors. Furthermore, taking into consideration interactions between monomers introduces a temperature dependence to the static properties of polymers that is not present with ideal chains.

2.2.3.1 Monomer Interactions and Excluded Volume

One key difference between ideal chains and real polymers is the emergence of *excluded volume*, that changes the statistics of the chain from a *random walk* to a *self-avoiding walk*. In ideal chains the monomers that are far apart along the chain do not interact and therefore do not influence each other, so that multiple monomers can occupy the same volume simultaneously. In real chains, the interactions between the monomers alter this behavior and typically result in an effective volume for every monomer, that is inaccessible to all other monomers. A typical approximation for the interaction between the monomers is a homogeneous, additive pair potential, that effectively takes into account both the energy cost of steric repulsion between two overlapping monomers as well as other energetic contributions coming from the monomer-monomer interactions and monomer-solvent interactions. Due to this, this effective interaction typically has a repulsive hard-core barrier combined with either an attractive well if it is energetically favorable for monomers to be close to monomers instead of -possibly implicit- solvent particles or with an extra repulsion if monomer-solvent contacts are more favorable than monomer-monomer contacts. If the solvent particles are chemically identical to the monomers, there is no difference in the energy between the different species and the effective potential often only consists of the repulsive hard-core barrier exclusively. The Lennard-Jones potential in Eq. (2.2.1.3) is an example for a combination of a hard-core repulsion with an attractive well.

The probability of finding two monomers at a certain distance is approximately³ proportional to the Boltzmann factor associated with this pairwise interaction.

For such a commonly assumed homogeneous, pairwise potential $U(\mathbf{r})$ the probability of finding those two particles at a distance r is therefore proportional to $e^{-\beta U(\mathbf{r})}$, with $\beta = \frac{1}{k_B T}$. The case of non-interacting particles is a special case of this with $U(\mathbf{r}) = 0 \forall \mathbf{r}$, yielding a Boltzmann factor of $e^{-\beta U(\mathbf{r})} = 1$ for all distances, indicating an uniform probability distribution, as one would expect.

³This approximation takes into account only pairwise correlations and neglects contributions due to all the other surrounding monomers.

This motivates the definition of the *Mayer- f -function*, which reads

$$f(\mathbf{r}) = e^{-\beta U(\mathbf{r})} - 1. \quad (2.2.3.1)$$

The Mayer- f -function can be interpreted as the excess probability weight of two non-interacting monomers being separated by a distance r compared to the non-interacting case. By integrating the Mayer- f -function over the whole space, *i.e.*

$$v = - \int f(\mathbf{r}) d^3\mathbf{r} \quad (2.2.3.2)$$

$$= \int 1 - e^{-\beta U(\mathbf{r})} d^3\mathbf{r} \quad (2.2.3.3)$$

one obtains the *net* two-body interaction between the monomers. The parameter v is called the *excluded volume* of the monomers and its sign indicates whether there is a net attraction ($v < 0$) or a net repulsion ($v > 0$) between the particles. If the potential U is radially symmetric, the excluded volume can be obtained as

$$v = -4\pi \int_0^\infty f(r) dr \quad (2.2.3.4)$$

$$= 4\pi \int_0^\infty 1 - e^{-\beta U(r)} dr. \quad (2.2.3.5)$$

This expression is related to the Barker-Henderson diameter,¹⁶ which is defined as

$$d_{\text{BH}} = \int_0^{\sigma_0} 1 - e^{-\beta U(r)} dr \quad (2.2.3.6)$$

with σ_0 being the distance for which the potential vanishes, $U(\sigma_0) = 0$. From comparison it becomes clear that the Barker-Henderson diameter is a truncated version of the integral in Eq. (2.2.3.5), measuring the size of the hard-core of the particles.

2.2.3.2 Solvent Qualities and Theta Temperature

As alluded to in the previous section, the pair interaction between monomers is both a consequence of the interactions between the monomers themselves as well as that between the -again possibly implicit- solvent particles. Hence, the excluded volume, that summarizes the net two-body interactions, can be used to distinguish five different types of solvent qualities:⁴

⁴Here we assume spherical Kuhn monomers with Kuhn length b_k .

1. For interactions that solely consist of hard-core repulsion, *e.g.* when the solvent particles and the monomers are chemically identical, the excluded volume becomes independent of the temperature, which is why the solvent is called *athermal*. In this case, $v \geq b_k^3$ and the statistics of the chains are that of a self-avoiding walk, *i.e.* a random walk where monomers are not allowed to overlap.
2. For *good solvents* there is a comparably small attractive contribution to the potential energy, leading to $0 < v < b_k^3$. In this case, the chain behaves like a random walk on scales smaller than a certain length determined by the Kuhn length of the monomers and the excluded volume, *cf.* Sec. 2.2.3.3. On larger length scales, the chain is swollen, due to the net repulsion between monomers, following the statistics of a self-avoiding walk.
3. For a *theta solvent* the attractive and repulsive contributions of the potential cancel each other out, yielding an excluded volume of $v = 0$. Due to this cancellation, there is no net two-body interaction, leading to the statistics of the chain to be approximately that of an ideal chain. Hence, a polymer in a theta solvent follows the statistics of a random walk.
4. For Polymers in a *poor solvent* the attraction of the pair interaction dominates the repulsion, resulting in a net attraction between monomers with the excluded volume being $-b_k^3 < v < 0$. Under these conditions the polymer again shows ideal behavior below the same length scale as for the good solvent, but collapses rather than swells beyond this length scale. On these larger length scales, the chain becomes a densely packed sphere of blobs, while on the smaller length scales the statistics again follows that of a random walk.
5. For *non-solvents* where $v \leq -b_k^3$ the attraction between monomers is so strong that the solvent is effectively excluded from the volume occupied by the chain. In this case, the chain is totally collapsed on all length scales, with the monomers densely filling the chain's volume.

As can be seen from Eqs. (2.2.3.1) and (2.2.3.2), in general the excluded volume depends on the temperature, which is why the solvent quality depends on the temperature as well. One distinguished temperature, called the *theta temperature* or θ -temperature is the temperature at which the excluded volume is exactly zero, $v = 0$. At the theta temperature, the monomers experience no net two-body interaction as measured by v and the corresponding chain consequently has nearly ideal conformations, leading to random walk statistics.

2.2.3.3 Scaling Behavior and Temperature Dependence

This section will provide a very concise introduction into the scaling behavior of chains in different solvent conditions. As the derivations would exceed the scope of this thesis, it will only sketch the principal ideas without going into further detail. Additional information can be found in the relevant literature, for example Ref. [17].

Let R denote the square root of the mean squared end-to-end distance,

$$R \equiv \sqrt{\langle (\mathbf{R}_{ee}^{\text{ideal}})^2 \rangle}, \quad (2.2.3.7)$$

then R is a measure for the typical size of the given polymer.

As derived in Sec. 2.2.2.1 and expressed explicitly in Eq. (2.2.2.52) the extension of an ideal chain as measured by the mean squared end-to-end distance is proportional to the number of segments. Hence, R is proportional to the square root of the number of segments.

$$R = b C_{\infty}^{\frac{1}{2}} (N - 1)^{\frac{1}{2}} \quad (2.2.3.8)$$

$$\propto b (N - 1)^{\frac{1}{2}}. \quad (2.2.3.9)$$

One consequence from *Flory theory*, a theory approximating the free energy of a polymer in a solvent, is that the size of a chain generally follows a power law,

$$R \propto b (N - 1)^{\nu}, \quad (2.2.3.10)$$

where the exponent ν depends on the quality of the solvent and is equal to the reciprocal value of the *fractal dimension*. For a chain following random walk statistics, *i.e.* an ideal chain or a chain in a theta solvent, this exponent becomes $\nu = \frac{1}{2}$. A self-avoiding walk has an observed scaling exponent of $\nu \approx 0.588$, while Flory theory predicts a value of $\nu = \frac{3}{5}$, which is surprisingly accurate given that it contains cancellations of errors stemming from the assumptions made for the entropy of an ideal chain and the interaction energy of an isolated real chain. Finally, a collapsed, *i.e.* closely packed, configuration has a scaling exponent of $\nu = \frac{1}{3}$.

In order to understand the temperature dependence of the scaling behavior of real chains, one typically divides the polymer into *thermal blobs*, each containing n_T monomers. Inside a thermal blob the energy contribution from all the excluded volume interactions is considered to be smaller than the thermal energy $k_B T$, so that the monomers inside this thermal blob behave like an ideal

chain with random walk statistics. The behavior of the collection of thermal blobs then depends on the excluded volume interactions. Hence, the size of the thermal blob, here denoted as ξ_T , defines a relevant length scale of the system. From the description above it becomes clear that the size of the thermal blobs depends on the excluded volume, which in turn depends on the temperature.

Let n_T denote the number of monomers in a thermal blob and ξ_T the size of the thermal blob. In Flory theory the average number of monomers being located in the excluded volume of a given monomer in a thermal blob is equal to the number of monomers times the ratio of the excluded volume and the pervaded volume, $n_T \frac{|v|}{\xi_T^3}$.⁵ For each overlap of monomers, an energetic penalty of $k_B T$ is assumed. As the expression above holds true for all n_T monomers in the blob, the cumulative interaction energy equals $n_T^2 \frac{|v|}{\xi_T^3} k_B T$.

As alluded to above, the cumulative interaction energy of a thermal blob is approximately equal to the thermal energy $k_B T$ and the chain inside of the thermal blob follows ideal random walk statistics, as the thermal energy is sufficient to make the effects from interactions between monomers negligible on scales smaller than the thermal blob. Hence, the governing equations of the size of a thermal blob and the number of monomers contained in it read

$$\xi_T \approx b_k n_T^{\frac{1}{2}} \quad (2.2.3.11)$$

$$n_T^2 \frac{|v|}{\xi_T^3} k_B T \approx k_B T \quad (2.2.3.12)$$

$$\Leftrightarrow n_T^2 \frac{|v|}{\xi_T^3} \approx 1. \quad (2.2.3.13)$$

By inserting Eq. (2.2.3.11) into Eq. (2.2.3.13) and solving for n_T , one finds

$$n_T^2 \frac{|v|}{\xi_T^3} \approx 1 \quad (2.2.3.14)$$

$$\Leftrightarrow n_T^2 \frac{|v|}{b_k^3 n_T^{\frac{3}{2}}} \approx 1 \quad (2.2.3.15)$$

$$\Leftrightarrow n_T^{\frac{1}{2}} \frac{|v|}{b_k^3} \approx 1 \quad (2.2.3.16)$$

$$\Leftrightarrow n_T \approx \frac{b_k^6}{|v|^2}. \quad (2.2.3.17)$$

⁵This approximation assumes that the positions of monomers are uncorrelated and that the monomers are evenly distributed in the pervaded volume of the chain, both of which are not the case in reality.

From the expression for n_T , Eq. (2.2.3.17), and the ideal statistics inside the thermal blob, Eq. (2.2.3.11), one can determine the blob's size,

$$\xi_T \approx \frac{b_k^4}{|v|}. \quad (2.2.3.18)$$

From Eqs. (2.2.3.17) and (2.2.3.18) one can see that both n_T and ξ_T diverge for $v \rightarrow 0$, so that for polymers in a theta solvent, the thermal blob size becomes infinite. This makes sense, as the chain shows ideal behavior on all length scales, whereby a thermal blob would comprise of all the monomers of the chain. If on the other hand $|v| \geq b_k^3$, then $n_T \leq 1$ and $\xi \leq b_k$, *i.e.* a thermal blob consists of only a single monomer and the thermal length scale is smaller or equal to the Kuhn length. This implies that for athermal solvents ($v \geq b_k^3$) and non-solvents ($v \leq -b_k^3$) there is no transition of the scaling behavior but they exhibit their respective scaling regardless of the number of segments they consist of.

The scaling behavior of real chains resulting from the considerations above is schematically summarized in Fig. 2.3 with both scales being logarithmic. In reality, the scaling behavior is more complex, in particular the transition from ideal random walk statistics below the thermal length scale ξ_T to self-avoiding walk statistics or collapsed chains is of course not as sharp as depicted in general.

2.3 Computer Simulations

Computer simulations in soft matter physics are a means of gaining insights into physical systems not via analytical calculations as in theory or by observing and measuring a real world system as in experiments but by numerically solving time or ensemble integrals. This is valuable due to fact that the properties one is interested in often involve very large systems and very complicated interactions which are far beyond what can be calculated analytically. On the other hand, experiments are limited by the technical practicability of the synthesis of the system as well as the necessary measurements, while computer simulations allow the study of arbitrary models and measurements. However, computer simulations do not yield analytical expressions but only expectation values of observables and are further limited by the computational resources and the need for efficient algorithms, which is why it replaces neither theoretical calculations nor experiments. Instead, these approaches compliment each other, with computer simulations being a connective link in between theory

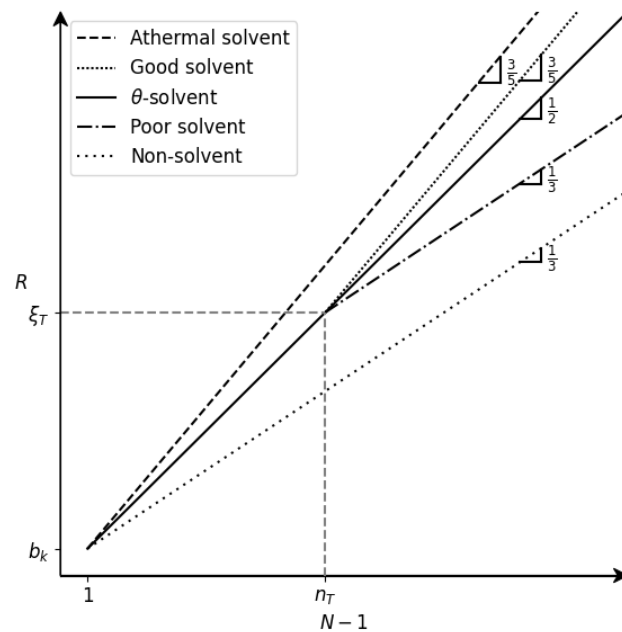


Figure 2.3: Log-log schematic of the scaling behavior of the size R of real polymers in different solvents as a function of the number of segments $(N - 1)$. For the self-avoiding walk statistics, the scaling exponent of $\nu = \frac{3}{5}$ from Flory theory was assumed. n_T and ξ_T denote the thermal length scale and the number of segments per thermal blob respectively. In reality, the scaling of the size of polymers with the number of segments is more complex, as the transition at (n_T, ξ_T) is not as sharp as depicted but typically smoother.

and experiments and hence representing a third paradigm of research in soft matter research.

In general, one distinguishes two types of computer simulations, *Monte Carlo* (MC) simulations and *molecular dynamics* (MD) simulations.

2.3.1 Monte Carlo Simulations

The principle idea behind MC has originally been proposed in 1777 by Georges-Louis Leclerc, Comte de Buffon in the famous “*Buffon’s needle problem*”,¹⁸ which can be used as a method to approximate π via a series of experiments. The method was then further developed and employed by different scientists until its modern version, the *Metropolis Monte Carlo method*, was published by Stanislaw Ulam *et al.*¹⁹ in 1949 as part of research aimed at the development of the nuclear bomb at Los Alamos National Laboratory. In the following, we will refer to the special case of the Metropolis Monte Carlo simply as “Monte Carlo” or “MC” if not stated otherwise explicitly.

In MC simulations, one aims at approximating ensemble averages of target quantities by conducting a large set of experiments and computing the corresponding expectation values. In the case of soft matter systems this corresponds to generating a substantial number of configurations of the system and calculating the expectation value of the observable $A(\cdot)$ as the weighted average of the realizations of $A(\cdot)$, *cf.* Eq. (2.1.1.9). While this would, in principle, yield accurate results for an infinite number of samples, it is infeasible due to the intractable number of computations that would have to be done. Thus, it is significantly more efficient to sample the configurations with a stronger focus on those examples with a higher statistical weight. Let $\mathcal{P}_B(\cdot)$ denote the true underlying probability distribution of configurations as given by the canonical probabilities, Eq. (2.1.1.4), and $\mathcal{P}_S(\cdot)$ the probability distribution describing the sampling routine of the MC method, then one finds

$$\langle A \rangle = \sum_{\mathcal{C}} \mathcal{P}_B(\mathcal{C}) A(\mathcal{C}) \quad (2.3.1.1)$$

$$= \sum_{\mathcal{C}} \mathcal{P}_B(\mathcal{C}) \frac{\mathcal{P}_S(\mathcal{C})}{\mathcal{P}_S(\mathcal{C})} A(\mathcal{C}) \quad (2.3.1.2)$$

$$= \sum_{\mathcal{C}} \mathcal{P}_S(\mathcal{C}) \frac{\mathcal{P}_B(\mathcal{C})}{\mathcal{P}_S(\mathcal{C})} A(\mathcal{C}) \quad (2.3.1.3)$$

$$\approx \frac{1}{|S|} \sum_{\mathcal{C} \sim \mathcal{P}_S} \frac{\mathcal{P}_B(\mathcal{C})}{\mathcal{P}_S(\mathcal{C})} A(\mathcal{C}) \quad (2.3.1.4)$$

as long as

$$\forall \mathcal{C}' \in \{\mathcal{C} \in \mathbf{P} | \mathcal{P}_B(\mathcal{C}) > 0\} : \quad \mathcal{P}_S(\mathcal{C}') > 0. \quad (2.3.1.5)$$

Here, \mathbf{P} stands for the phase space, *i.e.* the space of all possible microscopic configurations, and $\mathcal{C} \sim \mathcal{P}_S$ signifies that the configurations \mathcal{C} are sampled according to probability distribution \mathcal{P}_S . Furthermore, $S \subset \mathbf{P}$ denotes the set of samples and $|S|$ its cardinality, *viz.* the number of samples. The expression in Eq. (2.3.1.4) holds true approximately for a large but finite number of configurations so that by sampling and reweighting a subset of configurations according to the sampling probability distribution $\mathcal{P}_S(\cdot)$ one can approximate the expectation value of the physically correct probability distribution $\mathcal{P}_B(\cdot)$. In the case of these probability distributions being identical, *i.e.* $\mathcal{P}_B(\mathcal{C}) = \mathcal{P}_S(\mathcal{C}) \forall \mathcal{C} \in \mathbf{P}$, the expectation value becomes

$$\langle A \rangle \approx \frac{1}{|S|} \sum_{\mathcal{C} \sim \mathcal{P}_S} \frac{\mathcal{P}_B(\mathcal{C})}{\mathcal{P}_S(\mathcal{C})} A(\mathcal{C}) \quad (2.3.1.6)$$

$$= \frac{1}{|S|} \sum_{\mathcal{C} \sim \mathcal{P}_B} A(\mathcal{C}), \quad (2.3.1.7)$$

so in the case of sampling according to the physically correct probability distribution, the expectation value is obtained by an unweighted average of the observed realizations. The advantage of this lies in a more efficient way of sampling, as by sampling according to the physical distribution those examples with higher statistical weight will occur more frequently, while those which do not contribute as strongly to the expectation values will be sampled less often.

Metropolis Monte Carlo makes use of this observation by starting with a configuration and evolving it with trial moves, *i.e.* statistical modifications to the current configuration, and accepting the modification according to a chosen acceptance criterion. Here, the modifications can be chosen arbitrarily and do not have to follow any physical law. For example one might choose to move or rotate whole sets of particles like polymers or add or remove particles. This allows for exploring the phase space more efficiently, while other methods that are limited to physically feasible modifications tend to get stuck in local optima. The acceptance criterion has to be chosen in such a way, that the resulting transitions between configurations fulfill the criterion of *detailed balance*, which states, that the probability of being in state S_i , $\mathcal{P}(S_i)$, and transitioning to a state S_j is equal to the probability of being in state S_j , $\mathcal{P}(S_j)$, and transitioning to state S_i ,

$$\mathcal{P}(S_i) (\omega)_{ij} = \mathcal{P}(S_j) (\omega)_{ji} \quad (2.3.1.8)$$

where ω is the transition matrix with its components $(\omega)_{\alpha\beta}$ describing the probabilities of transitioning from state α to state β . The detailed balance condition implies the *global balance condition*,

$$\mathcal{P}(S_i) \sum_{j \neq i} (\omega)_{ij} = \sum_{j \neq i} \mathcal{P}(S_j) (\omega)_{ji} \quad (2.3.1.9)$$

which is a necessary and sufficient criterion for a stationary probability distribution of the states.²⁰ Furthermore, the detailed balance condition is often easier to prove than the global balance condition, while still being a sufficient condition.

A typical choice for the acceptance criterion that fulfills the detailed balance condition is the *Metropolis acceptance ratio*, where a trial move from state S_i to state S_j is accepted with probability

$$\text{Acc}(S_i, S_j) = \min(1, \exp[-\beta(\mathcal{H}(S_j) - \mathcal{H}(S_i))]). \quad (2.3.1.10)$$

Besides having to fulfill the global balance condition, the trial moves and associated acceptance probabilities have to be chosen such that the system is ergodic, as otherwise the configurational integral is effectively “truncated”, meaning not all possible states of the system are accounted for when calculating the expectation values of observables, possibly leading to incorrect results.

Despite standard MC methods not reproducing physical trajectories and therefore being incapable of predicting values of dynamical properties such as viscosities or diffusion coefficients, they are very efficient in calculating expectation values for static properties, like n -particle distribution functions and density profiles to name but a few examples. Due to the missing restrictions of the trial moves, it is possible to reduce the risk of getting stuck in local optima of the free energy landscape while exploring the phase space and it is straightforward to switch to different ensembles by adding corresponding trial moves and adjusting the acceptance probability. This is why even with increasing availability of high-performance computers and the advent of molecular dynamics simulations, new MC methods are still developed and frequently used in modern research.

2.3.2 Molecular Dynamics Simulations

While MC methods are designed to calculate the ensemble average of a property, with molecular dynamics (MD) simulations one obtains the time average of said property. Under the assumption of ergodicity, these two approaches

yield equal results for expectation values, according to Eq. (2.1.4.1). Hence, in these cases both methods can be used to calculate these values, depending on whichever is more suited for the system at hand. The idea behind molecular dynamics simulations is to evolve a system according to its equations of motion, *i.e.* by applying time-discretized versions of Newton's equation, $\mathbf{F} = m\mathbf{a}$, to the particles in the system as specified by the interaction Hamiltonian \mathcal{H} . MD simulations were devised in the 1950s with pioneering work such as that by Alder and Wainwright in 1957 when they published results of simulations with several hundred hard spheres.²¹ With many early successes in the application of MD simulations to more realistic systems like the calculation of the self-diffusion coefficient of liquid argon using a Lennard-Jones potential,²² *cf.* Eq. (2.2.1.3), this technique quickly became popular and established itself as an important tool for studying soft matter and for solid state research. Especially the inherent idea of mimicking the dynamics of the system as closely as possible proves very useful, as this allows for the calculation of dynamic properties like viscosities or diffusion coefficients.

2.3.2.1 Integrators in MD Simulations

Typically, the time evolution in MD simulations is done by alternately calculating the forces acting on the particles at the current time t and updating their velocities and positions accordingly using a *timestep* Δt , which is then done iteratively for a set number of cycles. While the interaction Hamiltonian \mathcal{H} is determined by the interactions of the particles in the system and hence can not be changed without skewing its properties, the choice of the update scheme, also known as *integrator*, is in principle arbitrary as long as it discretizes Newton's equations of motion. As it turns out, fulfilling Newton's equations of motion is not a sufficient criterion for the conservation of energy, momentum and angular momentum due to the discretization error, which is why the simplest integration scheme, the *Euler scheme*, is not an advisable integrator. The Euler scheme is defined as the truncated Taylor expansion of the current position of particle i , $\mathbf{r}_i(t)$, up to second order

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\mathbf{F}_i(t)}{2m_i} (\Delta t)^2 \quad (2.3.2.1)$$

with particle velocity $\mathbf{v}_i(t)$, mass m_i and $\mathbf{F}_i(t) = -\frac{\partial \mathcal{H}(\mathbf{r}(t))}{\partial \mathbf{r}_i(t)}$ being the force acting on particle i , where $\mathbf{r}(t)$ is the set of all positions at time t . Here, we have assumed that the interaction Hamiltonian does not depend on the set of current velocities, $\mathbf{v}(t)$. The Euler scheme has a local truncation error of the order of $\mathcal{O}((\Delta t)^2)$ and can be numerically unstable, *viz.* the deviation from the exact solution of the equations of motion can grow very large, which leads to violations

of the aforementioned conservation laws.

A sufficient criterion for the approximate conservation of said properties is the *symplecticity* of the integrator. A symplectic mapping is a linear transformation $A : \mathbb{R}^{2n} \rightarrow \mathbb{R}^{2n}$, with corresponding matrix \mathbf{A} , such that for every pair of vectors $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^{2n}$

$$\mathbf{z}_1^T \mathbf{J}_{2n \times 2n} \mathbf{z}_2 = (A(\mathbf{z}_1))^T \mathbf{J}_{2n \times 2n} A(\mathbf{z}_2) \quad (2.3.2.2)$$

$$= (\mathbf{A}\mathbf{z}_1)^T \mathbf{J}_{2n \times 2n} (\mathbf{A}\mathbf{z}_2) \quad (2.3.2.3)$$

$$= \mathbf{z}_1^T \mathbf{A}^T \mathbf{J}_{2n \times 2n} \mathbf{A} \mathbf{z}_2 \quad (2.3.2.4)$$

$$= \mathbf{z}_1^T (\mathbf{A}^T \mathbf{J}_{2n \times 2n} \mathbf{A}) \mathbf{z}_2 \quad (2.3.2.5)$$

$$\Leftrightarrow \mathbf{A}^T \mathbf{J}_{2n \times 2n} \mathbf{A} = \mathbf{J}_{2n \times 2n} \quad (2.3.2.6)$$

where the superscript T denotes transposition and $\mathbf{J}_{2n \times 2n}$ is the $(2n \times 2n)$ -dimensional symplectic unit matrix,

$$\mathbf{J}_{2n \times 2n} = \begin{pmatrix} \mathbf{0}_{n \times n} & \mathbb{1}_{n \times n} \\ -\mathbb{1}_{n \times n} & \mathbf{0}_{n \times n} \end{pmatrix}. \quad (2.3.2.7)$$

The mappings that fulfill Eq. (2.3.2.6) form the so-called *symplectic group* of dimension $2n$. Nonlinear differentiable mappings are also called symplectic, if their Jacobian matrix fulfills Eq. (2.3.2.6).

The symplectic structure is exactly that of Hamilton's equations,

$$\frac{d\mathbf{p}}{dt} = \frac{\partial \mathcal{H}(\{\mathbf{p}, \mathbf{q}\})}{\partial \mathbf{q}}, \quad \frac{d\mathbf{q}}{dt} = -\frac{\partial \mathcal{H}(\{\mathbf{p}, \mathbf{q}\})}{\partial \mathbf{p}} \quad (2.3.2.8)$$

with the vectors of all particle positions $\mathbf{p} \in \mathbb{R}^{\mathcal{N}}$ and particle momenta $\mathbf{q} \in \mathbb{R}^{\mathcal{N}}$, where \mathcal{N} denotes the number of particles, as it can be written as

$$\begin{pmatrix} \frac{d\mathbf{p}}{dt} \\ \frac{d\mathbf{q}}{dt} \end{pmatrix} = \mathbf{J}_{2\mathcal{N} \times 2\mathcal{N}} \nabla_{2\mathcal{N}} \mathcal{H}(\{\mathbf{p}, \mathbf{q}\}) = \mathbf{J}_{2\mathcal{N} \times 2\mathcal{N}} \begin{pmatrix} \frac{\partial \mathcal{H}(\{\mathbf{p}, \mathbf{q}\})}{\partial \mathbf{p}} \\ \frac{\partial \mathcal{H}(\{\mathbf{p}, \mathbf{q}\})}{\partial \mathbf{q}} \end{pmatrix} \quad (2.3.2.9)$$

with the $2\mathcal{N}$ -dimensional gradient $\nabla_{2\mathcal{N}}$.

Hence, integrator schemes have to be symplectic so that Eq. (2.3.2.9) is invariant under the associated transformation in order for them to fulfill Hamilton's equations. By virtue of *Liouville's theorem*²³⁻²⁵ fulfillment of these equations implies conservation of phase space volume and thus of energy and momentum.

One simple and often used scheme that is symplectic is the *Velocity Verlet* scheme²⁶ which reads

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\mathbf{F}_i(t)}{2m_i} (\Delta t)^2 \quad (2.3.2.10)$$

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \frac{\mathbf{F}_i(t) + \mathbf{F}_i(t + \Delta t)}{2m_i} \Delta t \quad (2.3.2.11)$$

where it is assumed again, that the force $\mathbf{F}_i(t) = -\frac{\partial \mathcal{H}(\mathbf{r}(t))}{\partial \mathbf{r}_i(t)}$ is independent of the set of particle velocities $\mathbf{v}(t)$. This integrator has a local truncation error of the order of $\mathcal{O}((\Delta t)^4)$, which is significantly better than that of the Euler scheme, which is why in the work presented in this thesis, we used this integration scheme.

2.3.2.2 Thermostats in MD Simulations

In order to transform the microcanonical \mathcal{NVE} ensemble to the canonical \mathcal{NVT} ensemble in MD simulations, there exist multiple variations of *thermostats*. These thermostats mimic the exchange of heat with an external bath, typically by modifying the particle velocities or the equations of motion. The easiest example of such a thermostat is the *Berendsen thermostat*,²⁷ which rescales the velocities of all particles, such that the *equipartition theorem*

$$\sum_{i=1}^{\mathcal{N}} \frac{1}{2} m_i \mathbf{v}_i^2 = \frac{3\mathcal{N}}{2} k_B T \quad (2.3.2.12)$$

is fulfilled. The Berendsen thermostat does not produce the correct Maxwell-Boltzmann distribution of velocities and therefore leads to incorrect results for static and dynamic properties, but is useful especially for equilibration of systems due to its simplicity.

A very commonly used thermostat that leads to the velocities being distributed according to the Maxwell-Boltzmann distribution is the *Langevin thermostat*,^{28,29} which was devised by Paul Langevin in order to qualitatively describe the Brownian motion of particles. In the Langevin equation, Newton's equations of motion is modified in such a way, that the exchange with the heat bath is implemented via random collisions and friction. Here, the collisions are modeled via an uniform random force $\boldsymbol{\eta}_i(t)$ with mean zero $\langle \boldsymbol{\eta}_i(t) \rangle = \mathbf{0}$, while the friction $\mathbf{F}_i^{\text{frict}}$ is given by a friction coefficient γ multiplied by the current velocity yielding,

$$m_i \frac{d\mathbf{v}_i}{dt} = \mathbf{F}_i(t) + \boldsymbol{\eta}_i(t) - \gamma \mathbf{v}_i(t). \quad (2.3.2.13)$$

The autocorrelation of the random force reads

$$\left\langle (\boldsymbol{\eta}_i(t))_\alpha (\boldsymbol{\eta}_j(\tau))_\beta \right\rangle = 2\gamma k_B T \delta_{ij} \delta_{\alpha\beta} \delta(t - \tau) \quad (2.3.2.14)$$

with α and β denoting the spatial dimension. The prefactor in Eq. (2.3.2.14) is chosen in such a way that the *fluctuation dissipation theorem* is fulfilled.

From the above description and Eq. (2.3.2.13) one can see, that Langevin's equations of motion are a *stochastic* partial differential equation instead of a *deterministic* partial differential equation like Newton's equations of motion.

In the case of the generalized Langevin equation,^{30,31} the friction term $\mathbf{F}_i^{\text{frict}}$ is described by a convolution of a *memory kernel* $\mathcal{M}(\cdot)$ with the history of previous velocities,

$$\mathbf{F}_i^{\text{frict}}(t) = \int_0^t \mathcal{M}(t - t') \mathbf{v}(t') dt'. \quad (2.3.2.15)$$

accounting for effects like backflow.

While the generalized Langevin equation is subject of ongoing research,^{32,33} in most cases one assumes that the movement of the particles in the system is on a much larger timescale than the movement of the particles from the implicit heat bath. In this case the memory kernel and consequently the friction term approximately become δ -correlated and therefore independent of the history of velocities as well as other particles' velocities,

$$\mathbf{F}_i^{\text{frict}}(t) = -\gamma \int_0^t \delta(t - t') \mathbf{v}_i(t') dt' \quad (2.3.2.16)$$

$$= -\gamma \mathbf{v}_i(t) \quad (2.3.2.17)$$

resulting in the expression in Eq. (2.3.2.13). The simulations in the \mathcal{NVT} ensemble in this thesis were conducted using the Langevin equation with such a δ -correlated friction term as described in Eq. (2.3.2.13).

2.3.3 Periodic Boundary Conditions

Typically, the number of particles that can realistically be simulated in MD for sufficient times is strongly limited by the complexity of the interaction Hamiltonian and the computational resources. Even with large computational resources it is often not possible to analyze the behavior of bulk systems, *i.e.* systems where the surface area of the phases is negligible compared to the volume of the phases. This is why in computer simulations one often employs a

technique called *periodic boundary conditions* (PBC), where particles are placed in a -often cubic- box with volume

$$V = (L_x \times L_y \times L_z) \quad (2.3.3.1)$$

with L_α denoting the size of the box in the direction $\alpha \in \{x, y, z\}$. The particles can interact with mirror particles outside the simulation box as if the mirror particles were independent of the particles inside the simulation box. In Fig. 2.4 we illustrate this by an example of two interacting particles in a two dimensional box.

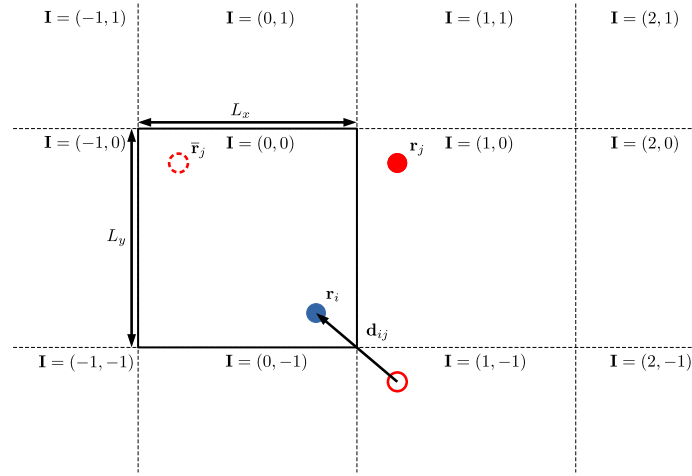


Figure 2.4: Illustration of two particles in a two dimensional simulation box with periodic boundary conditions. The images \mathbf{I} of the simulation box (solid line) and the mirror boxes (dashed line) can be found at the top of the corresponding boxes. Filled circles denote the absolute positions \mathbf{r}_i and \mathbf{r}_j , the empty circle with dashed contour corresponds to the position inside the box $\bar{\mathbf{r}}_i$ and the empty circle with solid contour indicates the virtual position of j for the calculation of the minimal distance d_{ij} between particles i and j , cf. Eq. (2.3.3.3).

In PBC, it makes sense to express the absolute position of a particle, \mathbf{r}_i , in terms of its position inside the box, $\bar{\mathbf{r}}_i$, and multiples of the box dimensions, called the *image* $\mathbf{I}_i \in \mathbb{N}^3$,

$$\mathbf{r}_i = \bar{\mathbf{r}}_i + \begin{pmatrix} (\mathbf{I}_i)_x L_x \\ (\mathbf{I}_i)_y L_y \\ (\mathbf{I}_i)_z L_z \end{pmatrix}. \quad (2.3.3.2)$$

Furthermore, the corresponding calculation of distances between particles i and j in a box is then modified from a regular euclidean distance to the *minimum*

distance convention,

$$\mathbf{d}_{ij} = \min_{\Delta \mathbf{I} \in \mathbb{N}^3} \left[\mathbf{r}_i - \mathbf{r}_j + \begin{pmatrix} (\Delta \mathbf{I})_x L_x \\ (\Delta \mathbf{I})_y L_y \\ (\Delta \mathbf{I})_z L_z \end{pmatrix} \right] \quad (2.3.3.3)$$

$$\equiv \min_{\Delta \mathbf{I} \in \mathbb{N}^3} \left[\bar{\mathbf{r}}_i - \bar{\mathbf{r}}_j + \begin{pmatrix} (\Delta \mathbf{I})_x L_x \\ (\Delta \mathbf{I})_y L_y \\ (\Delta \mathbf{I})_z L_z \end{pmatrix} \right] \quad (2.3.3.4)$$

where the minimum over $\Delta \mathbf{I}$ leads to the calculation of the distance of particle i with the *closest mirror particle of particle j* . Thus, the interactions between particles can cross the boundaries given by the simulation box, so that the simulation is effectively conducted on a higher dimensional torus.

2.3.4 Local Density-Dependent Potentials

In this thesis we employ a type of potential, which is not yet commonly used in soft matter research, the *local density-dependent potential* (LDP). The idea behind LDPs originates from the *embedded atom method* in solid state physics devised by Daw and Baskes.^{34,35} In their work, they derived a model of metals and impurities, where the interaction energy at impurities is not given by additive contributions but by an embedding energy that depends on the electron density at the corresponding positions.

Due to reasons of simplicity and because it is typically a valid assumption, the interaction Hamiltonian \mathcal{H} is often assumed to be a sum of additive n -body potentials,

$$\mathcal{H}(\mathbf{r}) = \frac{1}{2!} \sum_i \sum_{j \neq i} U^{2b}(\mathbf{r}_i, \mathbf{r}_j) + \frac{1}{3!} \sum_i \sum_{j \neq i} \sum_{k \notin \{i,j\}} U^{3b}(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) + \dots \quad (2.3.4.1)$$

$$\equiv \sum_i U_i(\mathbf{r}) \quad (2.3.4.2)$$

where the superscript of U denotes the number of particles participating in the corresponding interaction. Here, the potential energy $U_{i_1}^{nb}(\mathbf{r})$ of a particle i_1 due to the conventional n -body potential $U^{nb}(\mathbf{r})$ given the set of positions \mathbf{r} can therefore be expressed as

$$U_{i_1}^{nb}(\mathbf{r}) = \frac{1}{n!} \sum_{i_2 \neq i_1} \dots \sum_{i_n \notin \{i_1, i_2, \dots, i_{n-1}\}} U^{nb}(\mathbf{r}_{i_1}, \dots, \mathbf{r}_{i_n}). \quad (2.3.4.3)$$

The prefactors of $\frac{1}{n!}$ correct for counting sets of particles multiple times and assume that the potential functions are symmetric with regards to permutations of the corresponding particles.

In contrast, in the case of the local density-dependent approach a possibly nonlinear function, the *embedding function* G^{nb} , is applied to some additive tuple-wise *weight functions* $\omega^{nb}(\mathbf{r}_{i_1}, \dots, \mathbf{r}_{i_n})$. Thus the n -body local density-dependent potential of particle i_1 can be written as

$$U_{i_1}^{nb\text{-LDP}} = G^{nb} \left(\frac{1}{n!} \sum_{i_2 \neq i_1} \cdots \sum_{i_n \notin \{i_1, i_2, \dots, i_{n-1}\}} \omega^{nb}(\mathbf{r}_{i_1}, \dots, \mathbf{r}_{i_n}) \right) \quad (2.3.4.4)$$

$$= G^{nb} (\varphi_{i_1}^{nb}(\mathbf{r})). \quad (2.3.4.5)$$

where we have defined the n -body local density of particle i_1 , $\varphi_{i_1}^{nb}(\mathbf{r})$, as

$$\varphi_{i_1}^{nb}(\mathbf{r}) = \frac{1}{n!} \sum_{i_2 \neq i_1} \cdots \sum_{i_n \notin \{i_1, i_2, \dots, i_{n-1}\}} \omega^{nb}(\mathbf{r}_{i_1}, \dots, \mathbf{r}_{i_n}) \quad (2.3.4.6)$$

and the total energy due to the local density-dependent interactions reads

$$U^{\text{LDP}} = \sum_i U_i^{nb\text{-LDP}} \quad (2.3.4.7)$$

$$= \sum_i G^{nb} (\varphi_i^{nb}(\mathbf{r})). \quad (2.3.4.8)$$

Hence, from comparing Eq. (2.3.4.3) to Eq. (2.3.4.4) it becomes clear that the n -body LDP is a straight generalization of the conventional n -body potential, with the latter recovered from the former by choosing for example $\omega^{nb} = U^{nb}$ and G^{nb} as identity. Furthermore, as the number of applications of the embedding function G is negligible compared to the number of operations in the sums of additive terms, the computational complexity of the n -body LDP is the same as that of the conventional n -body potential.

However, while computationally equally demanding as its conventional counterpart, the n -body LDP can incorporate multi-body contributions to the potential energy of particle i , as it does not only depend linearly on the positions of other particles, but on all positions simultaneously, which define the aforementioned n -body local density $\varphi_i^{nb}(\mathbf{r})$. Thus, the particle's interaction is sensitive to its local environment, which is why it is especially useful in the application in inhomogeneous systems, as will be shown for the case of two-body

LDPs in Ch. 4.

In previous works, for example by the group of Scott Shell,^{36,37} the local density φ_i is typically defined such that it effectively counts the number of particles in the vicinity of a particle i , *e.g.* using a smoothed Heaviside function. Therefore, the corresponding LDP effectively introduces an additional cohesion between particles. The work presented in Ch. 4 and Ch. 5 differs from that in that the definition of the local density is instead given by a structural quantity obtained from microscopic references and is supposed to allow for a continuous transition from an effective pair potential at low density to one at the bulk density via inclusion of multibody effects.

2.3.5 Reduced Units

In computer simulations it is quite common to use *reduced units* rather than *real units* to express physical quantities like temperatures, energies, densities, pressures, *etc.* As the parameters of simulations and physical quantities can be quantified in arbitrary systems of units, one typically chooses a convenient set of basic units, in which these variables are expressed. From these basic units, other units may then be derived. The choice of these basic units is obviously not unique, as one may use different reference scales. In the case of a system of particles of mass m interacting via a Lennard-Jones potential (*cf.* Eq. (2.2.1.3)) it often makes sense to use its parameters as reference scale,

- σ as unit of length
- ε as unit of energy
- m as unit of mass.

From this choice of reference scales, one can derive other quantities, *e.g.* the unit of time as

$$[t] = \sigma \sqrt{m/\varepsilon} \quad (2.3.5.1)$$

while the temperature is expressed in terms of the interaction strength,

$$[T] = \varepsilon/k_B. \quad (2.3.5.2)$$

Quantities given in reduced units are then dimensionless and often denoted by a superscript “*”, for example

$$t^* = t / \left(\sigma \sqrt{m/\varepsilon} \right) \quad \text{and} \quad T^* = T / (\varepsilon/k_B) = k_B T / \varepsilon. \quad (2.3.5.3)$$

| Symbol | Parameter description | Unit |
|----------|-----------------------|------------------------------|
| l | Length | σ |
| E | Energy | ε |
| M | Mass | m |
| T | Temperature | ε/k_B |
| t | time | $\sigma\sqrt{m/\varepsilon}$ |
| ρ | Density | $1/\sigma^3$ |
| ρ | Force | ε/σ |
| k | Spring constant | ε/σ^2 |
| γ | Surface tension | ε/σ^2 |
| P | Pressure | ε/σ^3 |

Table 2.2: Overview of basic reduced units and units derived from these.

In Tab. (2.2) we summarize the basic units and some derived units used in this thesis.

Besides providing a more convenient unit system, this convention also allows for easier mapping of one system in reduced units to infinitely many systems in real units, also known as *law of corresponding states*.^{38,39} For example, one simulation of a Lennard-Jones model in reduced units might be mapped to a system consisting of Argon at $T = 60\text{K}$ and a density of $\rho = 840 \frac{\text{kg}}{\text{m}^3}$ as well as to a system consisting of Xenon at $T = 112\text{K}$ and a density of $\rho = 1617 \frac{\text{kg}}{\text{m}^3}$, both of which correspond to a simulation at $\rho^* = 0.5$ and $T^* = 0.5$.⁴⁰ In the first case, the energy scale would then for example be given as $\varepsilon \approx 0.499 \frac{\text{kJ}}{\text{mol}}$ and in the second as $\varepsilon \approx 0.931 \frac{\text{kJ}}{\text{mol}}$.

Additionally, using reduced units has the advantage of many quantities being expressible at the order of about 1, which is advantageous in numerical simulations, as operations using quantities differing by multiple orders of magnitude might introduce numerical inaccuracies due to the finite precision of numbers on computers.

Typically, one omits the superscript “*” for variables in reduced units and implicitly expresses all quantities in reduced units. This thesis follows this convention if not stated otherwise explicitly.

2.4 Coarse-Graining

While in principle one could simulate all systems with great detail by numerically solving the quantum mechanical equations such as the Schrödinger equation with the corresponding time-dependent Hamiltonian, this is prohibitively time consuming, limiting the applicability of this approach to very small time and length scales with system sizes of the order of a few hundred to a few thousand particles. For simulations of larger systems, like in the case of protein simulations, the computational effort therefore has to be reduced systematically in such a way, that the relevant properties of the system remain as close as possible to the underlying system. As different properties are subject to different time and length scales, one needs modeling techniques covering and bridging multiple scales, which is why these techniques are referred to as *multiscale modeling*. Generally, one distinguishes two types of multiscale modeling techniques, namely *bottom-up* and *top-down* approaches. In bottom-up approaches, one reduces the level of detail of a more complex system and tries to reproduce the microscopic features, while in top-down approaches, one introduces some level of detail to a less complex system in order to reproduce the features on a larger scale. The different time and length scales as well as corresponding systems in multiscale modeling are depicted in Fig. 2.5.

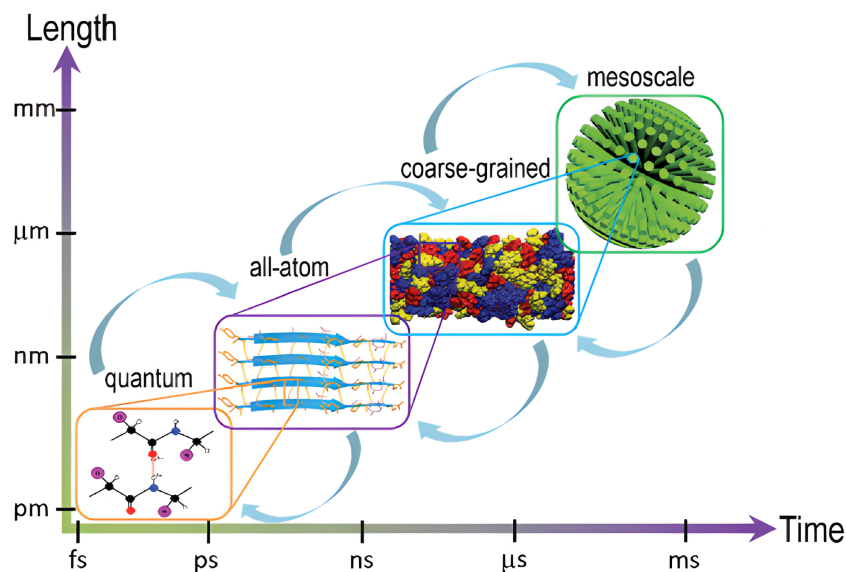


Figure 2.5: Illustration of multiscale modeling with different time and length scales and corresponding systems. Picture taken from Ref. [41].

One specific field of study in multiscale modeling is called *coarse-graining*,

which aims at the reduction of a model's complexity by reducing the degrees of freedoms such as the number of particles and by simplifying the interaction Hamiltonian. Coarse-graining is typically associated with a time and length scale inbetween atomistic systems, where every single atom is treated as a separate particle, and mesoscale systems, where mesoscopic structures are studied. Hence, on the scale of coarse-grained simulations usually multiple atoms are combined into single beads with effective interactions such that chosen properties of the microscopically resolved system are reproduced. In principle, the ideal model would reproduce all of these properties on the associated scales simultaneously, however, as the reduction of the model complexity is equivalent to a loss of information, which is irreversible, this is intrinsically not possible. In the following, there will be a discussion of three coarse-graining techniques used in this thesis, which aim at reproducing different features of the reference systems. While these techniques differ in the formulation of their corresponding optimization problem, they are intrinsically related as was discussed by Rudzinski and Noid.⁴²

2.4.1 Iterative Boltzmann Inversion

The technique called *Iterative Boltzmann Inversion*^{43,44} (IBI) is an example of *structure-based coarse-graining* as its objective is to obtain a pair potential which gives rise to the same radial distribution function, $g(r)$, cf. Sec. 2.1.2, as the reference system. The underlying idea of this approach is the *Henderson theorem*⁴⁵ which states that in systems with pairwise interactions the pair potential which gives rise to a certain RDF is unique up to a constant. This implies that if the RDF is matched perfectly, the pair potential is matched perfectly as well.

As the name suggests, IBI is an iterative method which alternates between conducting coarse-grained simulations and updating the pair potential $U(r)$ according to the difference of the RDF observed in iteration j , $g^j(r)$, and the reference one $g_{\text{ref}}(r)$. The update rule makes use of the low density approximation of $g(r)$, Eq. (2.1.2.15), and reads

$$U^{j+1}(r) = U^j(r) - \alpha k_{\text{B}}T \ln \left(\frac{g^j(r)}{g_{\text{ref}}(r)} \right) \quad (2.4.1.1)$$

where α is a scaling factor to prevent large fluctuations in the update. The initial guess of the pair potential, $U^0(r)$, is given by Eq. (2.1.2.15) as well and reads

$$U^0(r) = -k_{\text{B}}T \ln(g_{\text{ref}}(r)). \quad (2.4.1.2)$$

While there exist multiple similar but more sophisticated methods like the *Inverse Monte Carlo*⁴⁶ or the *hypernetted-chain* method^{47,48} this approach is still frequently used due to its simplicity and because it often yields good results. However, the method is limited by the locality of its update, as it does not consider the potential at distance r_1 potentially having an effect on the RDF at distance $r_2 \neq r_1$ in the update rule. Furthermore, it is very time consuming due to the need for simulations at every step in this iterative scheme, which is why in this thesis, we aim to make a contribution to faster structure-based coarse-graining using neural networks, *cf.* Ch. 3.

2.4.2 Force Matching

The *force matching*^{49–51} method aims at matching the *multibody potential of mean force* (PMF) instead of reproducing a distribution function. The Boltzmann factor for a configuration of n_{CG} coarse-grained beads $\mathbf{R} = \{\mathbf{R}_1, \dots, \mathbf{R}_{n_{\text{CG}}}\}$ is given by the Boltzmann weight as determined by the coarse-grained interaction potential $\mathcal{V}^{\text{CG}}(\mathbf{R})$. Simultaneously, it is given by the Boltzmann factors of the microscopic configurations $\mathbf{r} = \{\mathbf{r}_1, \dots, \mathbf{r}_n\}$ of n atomistic particles that map to \mathbf{R} according to the mapping \mathbf{M} , such that

$$e^{-\beta\mathcal{V}^{\text{CG}}(\mathbf{R})} = \int e^{-\beta\mathcal{H}(\mathbf{r})} \delta(\mathbf{M}(\mathbf{r}) - \mathbf{R}) \, d\mathbf{r}. \quad (2.4.2.1)$$

A typical choice of the mapping \mathbf{M} is to combine multiple particles into an effective bead located at the center of mass.

From Eq. (2.4.2.1) one immediately has an expression for the potential of mean force,

$$\mathcal{V}^{\text{CG}}(\mathbf{R}) = -k_{\text{B}}T \ln \left(\int e^{-\beta\mathcal{H}(\mathbf{r})} \delta(\mathbf{M}(\mathbf{r}) - \mathbf{R}) \, d\mathbf{r} \right). \quad (2.4.2.2)$$

However, as it requires the calculation of an integral over all microscopic configurations it is not feasible to compute the PMF using Eq. (2.4.2.2). Besides that, solving this integral to obtain the PMF would defeat the purpose of computer simulations as according to Eq. (2.1.1.8) the expectation values of all observables could be calculated directly instead of conducting simulations.

Instead, one resorts to approximately solving the *least-squares problem* to which the PMF is the analytical solution. This least-squares problem, also called *multiscale coarse-graining* (MS-CG) *residual*, consists of finding the coarse-grained forces \mathbf{F}^{CG} given a set of atomistic configurations \mathcal{C} and corresponding

atomistic forces $\mathbf{F}(\mathbf{r})$) such that the MS-CG residual

$$\chi^2 = \frac{1}{\dim(\mathbf{F}^{\text{CG}}) |\mathcal{C}|} \sum_{\mathbf{r} \in \mathcal{C}} \sum_{i=1}^{n_{\text{CG}}} |\mathbf{M}(\mathbf{F}(\mathbf{r}))_i - \mathbf{F}^{\text{CG}}(\mathbf{M}(\mathbf{r}))_i| \quad (2.4.2.3)$$

is minimized. Here, \mathbf{M} is again the mapping from atomistic to coarse-grained configurations and the subscript i denotes the force acting on the i -th CG bead. The dimensionality of the coarse-grained forces is then given by $\dim(\mathbf{F}^{\text{CG}}) = 3n_{\text{CG}}$ for a three dimensional system. In force matching the MS-CG residual in Eq. (2.4.2.3) is minimized numerically, typically by solving a system of linear equations, yielding an approximate PMF. In this thesis, we employed force matching to obtain an effective pair potential for two polymers in an empty box with the polymers being mapped to their respective centers of mass.

2.4.3 Relative Entropy Minimization

In *relative entropy minimization*^{36,37,52} (REM) the quantity to minimize is the *relative entropy* or *Kullback-Leibler divergence*^{53,54} between the atomistic and the coarse-grained probability distributions of configurations. Let \mathcal{Q} denote the probability distribution of configurations in the coarse-grained systems with the mapping function \mathbf{M} . Then one can define a corresponding probability function for configurations of the atomistic reference \mathcal{Q}' by accounting for the degeneracy of mapping an atomistic configuration to the CG configuration,

$$\mathcal{Q}'(\mathbf{r}) = \frac{\mathcal{Q}(\mathbf{M}(\mathbf{r}))}{\Omega(\mathbf{M}(\mathbf{r}))} \quad (2.4.3.1)$$

where the degeneracy Ω counts, how many atomistic configurations are mapped to the given CG configuration,

$$\Omega(\mathbf{R}) = \sum_{\mathbf{r} \in \mathcal{C}} \delta_{\mathbf{R}, \mathbf{M}(\mathbf{r})}. \quad (2.4.3.2)$$

Let further \mathcal{P} denote the probability distribution of the configurations in the atomistic reference system, then the relative entropy of these probability distributions is given as

$$S_{\text{rel}} = \sum_{\mathbf{r} \in \mathcal{C}} \mathcal{P}(\mathbf{r}) \ln \left(\frac{\mathcal{P}(\mathbf{r})}{\mathcal{Q}'(\mathbf{r})} \right) \quad (2.4.3.3)$$

$$= \sum_{\mathbf{r} \in \mathcal{C}} \mathcal{P}(\mathbf{r}) \ln \left(\frac{\mathcal{P}(\mathbf{r})}{\mathcal{Q}(\mathbf{M}(\mathbf{r}))} \right) + \sum_{\mathbf{r} \in \mathcal{C}} \mathcal{P}(\mathbf{r}) \ln(\Omega(\mathbf{M}(\mathbf{r}))). \quad (2.4.3.4)$$

The last term depends only on the mapping function and can therefore not be optimized when parametrizing the interaction potentials. Hence, minimizing the relative entropy corresponds to minimizing

$$S'_{\text{rel}} = \sum_{\mathbf{r} \in \mathcal{C}} \mathcal{P}(\mathbf{r}) \ln \left(\frac{\mathcal{P}(\mathbf{r})}{\mathcal{Q}(\mathbf{M}(\mathbf{r}))} \right). \quad (2.4.3.5)$$

For the canonical ensemble, one can use the canonical probabilities from Eq. (2.1.1.4) in Eq. (2.4.3.5) to derive

$$S'_{\text{rel}} = \beta \langle U - U_{\text{CG}} \rangle - \beta (A - A_{\text{CG}}) \quad (2.4.3.6)$$

where the average is taken in the atomistic system, U and U_{CG} denote the energy and A and A_{CG} the configurational part of the Helmholtz free energy in the microscopically resolved and the coarse-grained system respectively.

The residual defined in Eq. (2.4.3.6) can then be minimized by applying root-finding methods to its derivative. The method proposed by Scott Shell³⁶ and used in this thesis is the Newton-Raphson method, which yields the following update rule for the i -th parameter λ_i of the CG potential U_{CG} in the j -th iteration:

$$\lambda_i^{j+1} = \lambda_i^j - (\partial S'_{\text{rel}} / \partial \lambda_i) / (\partial^2 S'_{\text{rel}} / \partial \lambda_i^2) \quad (2.4.3.7)$$

$$= \lambda_i^j - \Delta \lambda_i^j \quad (2.4.3.8)$$

$$\begin{aligned} \Delta \lambda_i^j = & \left[\left\langle \frac{\partial U_{\text{CG}}}{\partial \lambda_i} \right\rangle_{\text{MR}} - \left\langle \frac{\partial U_{\text{CG}}}{\partial \lambda_i} \right\rangle_{\text{CG}} \right] \\ & \left[\left\langle \frac{\partial^2 U_{\text{CG}}}{\partial (\lambda_i)^2} \right\rangle_{\text{MR}} - \left\langle \frac{\partial^2 U_{\text{CG}}}{\partial (\lambda_i)^2} \right\rangle_{\text{CG}} + \beta \left\langle \left(\frac{\partial U_{\text{CG}}}{\partial \lambda_i} \right)^2 \right\rangle_{\text{CG}} - \beta \left\langle \frac{\partial U_{\text{CG}}}{\partial \lambda_i} \right\rangle_{\text{CG}}^2 \right]^{-1} \end{aligned} \quad (2.4.3.9)$$

where $\langle \cdot \rangle_{\text{MR}}$ corresponds to an average in the microscopically resolved simulation and $\langle \cdot \rangle_{\text{CG}}$ to an average in the coarse-grained system.

In Ch. 4 we used a modified version of this update rule with a step size parameter $\alpha = 0.01$ and a truncation at $\pm k_{\text{B}}T$, in order to ensure more stable updates, such that the modified update $\bar{\Delta} \lambda_i^j$ reads

$$\bar{\Delta} \lambda_i^j = \min(-1, \max(1, \alpha \Delta \lambda_i^j)). \quad (2.4.3.10)$$

Furthermore, in our work we used cardinal B-splines,⁵⁵ which are linear in the amplitude of their nodes, which yields a simplified update,

$$\Delta \lambda_i^j = k_{\text{B}}T \left[\left\langle \frac{U_{\text{CG}}}{\lambda_i} \right\rangle_{\text{MR}} - \left\langle \frac{U_{\text{CG}}}{\lambda_i} \right\rangle_{\text{CG}} \right] \left[\left\langle \frac{U_{\text{CG}}}{(\lambda_i)^2} \right\rangle_{\text{MR}} - \left\langle \frac{U_{\text{CG}}}{(\lambda_i)^2} \right\rangle_{\text{CG}} \right]^{-1} \quad (2.4.3.11)$$

with our modified version $\bar{\Delta} \lambda_i^j$ as before.

2.5 Machine Learning and Neural Networks

In this section, we will give a short introduction to the principle idea of machine learning as well as introduce the concepts used in the scope of this thesis, which includes a description of neural networks and their training procedure. As a more thorough and detailed introduction to machine learning in general is beyond the scope of this thesis, we refer to the relevant literature in this regard.^{56–58}

2.5.1 Machine Learning

Machine learning (ML) is an emergent field of study, that has had a lot of groundbreaking successes especially in the last decade. It has been applied to a huge variety of different problems, including everything ranging from computer vision and fraud detection to natural language processing and recommender systems, and is ubiquitous by now in our everyday life.

The central idea of machine learning is very well captured by a definition provided by Tom Mitchell,⁵⁶ which states:

A computer program is said to *learn* from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with experience E .

This means that machine learning is the study of solving problems by making use of large quantities of examples or interactions and optimizing the performance according to a chosen metric. One typically distinguishes three main types of machine learning, which differ in the variables mentioned above, namely the tasks T , the performance measure P and the experience E .

2.5.1.1 Categories of Machine Learning

In *supervised learning*, the task for the ML algorithm is to approximate a mapping $\Omega : \mathbf{X} \mapsto \mathbf{Y}$ from some input properties \mathbf{X} to some corresponding target properties \mathbf{Y} , also known as *ground truth* or *target*, with the experience provided by a set of N examples, $E = \{(\mathbf{X}_i, \mathbf{Y}_i) | i \in \{1, \dots, N\}\}$. If Θ denotes the parametrization of the ML model and $\Omega_\Theta : \mathbf{X} \mapsto \Omega_\Theta(\mathbf{X})$ the corresponding mapping, then the performance P on example (\mathbf{X}, \mathbf{Y}) is measured by applying a distance metric M to the output of the ML algorithm and the ground truth,

$$P_\Theta(\mathbf{X}) = M(\mathbf{Y}, \Omega_\Theta(\mathbf{X})). \quad (2.5.1.1)$$

The chosen metric typically depends on the desired mapping, with typical examples being regression tasks, where the target \mathbf{Y} is a real-valued vector, or classification tasks, where the target \mathbf{Y} corresponds to a vector of (possibly one-hot-encoded) labels. However, also other tasks and combinations of said tasks can occur, *e.g.* in object detection with bounding boxes, where the input \mathbf{X} comprises of image data or a map of pixel-wise distances and the desired output is a set of anchors, widths and heights describing boxes around different objects in the input.

In *unsupervised learning* the task for the machine learning algorithm is to approximate an identity mapping, $\Omega : \mathbf{X} \mapsto \mathbf{X}$ with \mathbf{X} providing both the input as well as the target, such that the experience is a set consisting of $E = \{\mathbf{X}_i | i \in \{1, \dots, N\}\}$. The performance on these tasks for a specific example \mathbf{X} is measured by a metric M between the input and the output,

$$P_{\Theta}(\mathbf{X}) = M(\mathbf{X}, \Omega_{\Theta}(\mathbf{X})). \quad (2.5.1.2)$$

By choosing the specifics of the algorithm in certain limiting ways, one can force the ML model to learn a lower dimensional intermediate representation of the data in the so-called *latent space*. This can be used for identifying and calculating order parameters or collective variables, as a low-dimensional representation of an input property, which carries sufficient information to be approximately invertible, must capture the essential abstract features of the input. Furthermore, a typical application of unsupervised learning is clustering analysis, where the goal is to find the underlying structure of data and cluster the data into distinct groups in the latent space depending on their abstract similarity.

Reinforcement learning is conceptually very different from supervised and unsupervised learning, as it does not necessarily rely on a given set of examples, but rather on the interaction with an environment, where in response to the interaction, the model obtains an immediate *reward*. Instead of predicting a certain target property provided by the training examples, the ML model is supposed to choose the best action a given a certain state \mathbf{S} , *i.e.* $\Omega : \mathbf{S} \mapsto \mathbf{a}$, where the best choice is often not known and depends *i.a.* on the transition in states that the chosen action causes. The performance of a model with parameters Θ in state \mathbf{S} is then typically measured by the expected discounted return $\langle \mathbf{G} \rangle_{\Theta}$ it obtains when following its *policy*, *viz.* for each state \mathbf{S}_i taking the action as proposed by the model $\mathbf{a}_i = \Omega_{\Theta}(\mathbf{S}_i)$ and obtaining rewards according to a reward function $R : (\mathbf{S}_i, \mathbf{a}_i) \mapsto r_i$,

$$P_{\Theta}(\mathbf{S}) = \langle \mathbf{G} \rangle_{\Theta} \quad (2.5.1.3)$$

$$= \left\langle \sum_{\mathbf{S}_i \in \mathcal{S}_\Theta} \gamma R(\mathbf{S}_i, \mathbf{a}_i) \right\rangle \quad (2.5.1.4)$$

with \mathcal{S}_Θ being the set of states visited by the model when following its policy. Hence, the experience E typically consists of examples seen during the interaction with the environment and therefore comprises of a history thereof, such that $E = \{(\mathbf{S}_i, \mathbf{a}_i, R(\mathbf{S}_i, \mathbf{a}_i))\}$.

The above descriptions are of course far from complete and do not cover a lot of important aspects and techniques but rather give a rough outline of these categories. For example, in the case of unsupervised learning, one might skew the performance measure by enforcing some structure to the latent space, or in the case of reinforcement learning one might look at non-deterministic but statistical policies, rewards and transitions between states, as well as different notions of reinforcement learning like *on-policy vs. off-policy* learning or *value-based vs. policy-based* methods.

While the categories mentioned above differ in their targets, caveats and technicalities, what unites these methods is that the respective goal is obtained by learning from examples and parametrizing the model in such a way, that the target is fulfilled as much as is feasible. Furthermore, all of these techniques have in common, that they approximate some - possibly abstract - function either implicitly or explicitly.

There is a plethora of different types of models being applied in machine learning, ranging from very simple ones like logistic regressions or k -nearest neighbor algorithms to more complicated ones like genetic algorithms or graph models. Another model, that is commonly used and has been applied very successfully especially in the past decade, is that of *artificial neural networks*, or *neural networks* for short, which will be discussed in more details in Sec. 2.5.2. In Fig. 2.6 we provide a Venn diagram illustrating the relations between some of the different terminologies used in artificial intelligence research.

We will restrict the discussion in Sec. 2.5.2 to a short introduction of the fundamental building blocks of the *feed-forward* neural networks used in this work, namely *fully-connected layers* as well as *convolutional layers*, and will briefly summarize the training procedure. For more detailed insights and an overview as well as the mathematical foundations of the numerous different architectures of neural networks employed today, we refer to the relevant literature, for example Ref. [57].

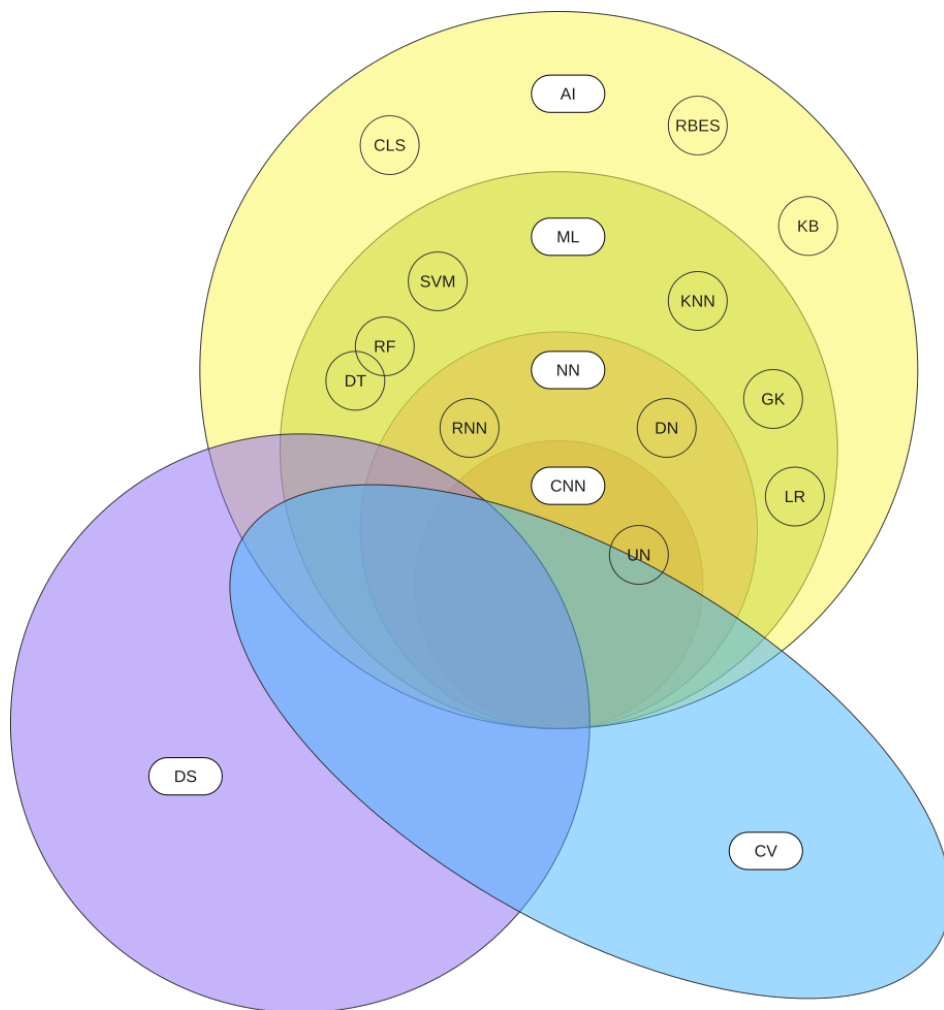


Figure 2.6: Venn diagram of terminologies in artificial intelligence research. Common names of fields are shown in white boxes, and typical examples of techniques in circles.

The abbreviations of fields for stand for: “AI” = “Artificial Intelligence”, “ML” = “Machine Learning”, “NN” = “Neural Networks”, “CNN” = “Convolutional Neural Networks”, “DS” = “Data Science”, “CV” = “Computer Vision”. The abbreviations of examples for stand for: “CLS” = “Closed Loop System”, “RBES” = “Rule Based Expert System”, “KB” = “Knowledge Base”, “DT” = “Decision Tree”, “RF” = “Random Forest”, “SVM” = “Support Vector Machine”, “KNN” = “ k -Nearest Neighbor”, “GK” = “Gaussian Kernel”, “LR” = “Linear Regression”, “UN” = “U-Net”.

2.5.1.2 Training, Validation and Testing

A very important concept that further unites all machine learning methods is the need to *train*, *validate* and *test* all models before deploying them, so that it is common to divide the available data into different sets for these purposes. While the training data is used to adjust the model parameters in order to optimize the objective, the validation data is used to detect possible overfitting of the ML model during training, *cf.* Sec. 2.5.3, and to serve as a first indication of the model's generalization capabilities. After training multiple ML models, one chooses a final model according to the performance on both the training set as well as the validation set. The test data is then used for a final benchmark of this model in order to get a realistic estimate of the generalization of the model's performance on previously unseen data. Ideally, the distributions in all data sets is similar and do not suffer from a *data shift*, as otherwise the aforementioned estimates are skewed. Furthermore, the data should of course be representative for the inputs the model will be confronted with in the real world, as otherwise the model's performance will almost certainly decrease and the model might even behave unpredictably.

While the separation between training data and validation data does not have to be very strict, for example in *cross-validation*, where the same set of training examples is used in different splits into training and validation sets, it is of utmost importance according to *best machine learning practices* to have a designated test set, that is only used for a final benchmark *once*. This is to avoid overfitting on the test set, which occurs when the model is chosen or adjusted according to the performance on this data set. Hence, it is crucial to keep this data set separate from the training and validation set and only use it in the final benchmark. We stress that in this work we followed the aforementioned best practices in machine learning.

2.5.2 Neural Networks

Neural networks (NNs) are inspired by the human brain in that the calculations in neural networks mimic the processing of signals in our neurons in a simplistic manner. Schematically, the signals in our body are passed from neuron to neuron, with all signals coming into a neuron being combined using weights according to the physical strength of the connection between the corresponding neurons and then passed on, if a certain threshold is exceeded. The weights of the connections, *i.e.* their conductivity, changes over time, with the connection being strengthened, if in retrospective the respective input proves relevant for a given task, and weakened otherwise. In our current understanding, this process

is deemed to be the underlying mechanism of *learning*, and is captured in the *Hebbian theory* in neuroscience, stating⁵⁹

Let us assume that the persistence or repetition of a reverberatory activity (or "trace") tends to induce lasting cellular changes that add to its stability. [...] When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.

often summarized as⁶⁰

Neurons wire together if they fire together.

This simplified description of the biological process is depicted on the left side of Fig. 2.7.

2.5.2.1 Artificial Neurons and Fully-Connected Layers

Mathematically, the accumulation of signals is realized as a weighted sum of the signals where the weights are adjustable parameters and the threshold in general is implemented via a nonlinear activation function. Hence, the *activation of a neuron* $z(\mathbf{X})$ given an m -dimensional input $\mathbf{X} \in \mathbb{R}^m$ can be written as

$$z(\mathbf{X}) = a \left(\sum_{i=1}^m [(\boldsymbol{\omega})_i(\mathbf{X})_i] + b \right) \quad (2.5.2.1)$$

$$= a(\boldsymbol{\omega} \cdot \mathbf{X} + b) \quad (2.5.2.2)$$

where $\boldsymbol{\omega} \in \mathbb{R}^m$ is the so-called *weight vector*, and $a : \mathbb{R} \rightarrow \mathbb{R}$ is the *nonlinear activation function*, or *activation function* for short. Furthermore, the parameter b is called *bias* and serves as a shift of the scalar product of the weight vector and the input vector, and thus functions akin to a learnable threshold parameter. This *artificial neuron* can be generalized to a higher dimensional output $\mathbf{z}(\mathbf{X}) \in \mathbb{R}^n$, so a set of n artificial neurons, by transforming $\boldsymbol{\omega}$ to a *weight matrix* instead of a weight vector and the bias b to a *bias vector* $\mathbf{b} \in \mathbb{R}^n$. Furthermore, the activation function $a : \mathbb{R} \rightarrow \mathbb{R}$ is applied component-wise, so that Eq. (2.5.2.1) becomes

$$(\mathbf{z}(\mathbf{X}))_i = a \left(\sum_{j=1}^m (\boldsymbol{\omega})_{ij}(\mathbf{X})_j + \mathbf{b}_i \right) \quad (2.5.2.3)$$

$$= a(\boldsymbol{\omega}_i \cdot \mathbf{X}) . \quad (2.5.2.4)$$

One typically does not mention that a is applied component-wise explicitly and just writes

$$\mathbf{z}(\mathbf{X}) = a(\boldsymbol{\omega} \cdot \mathbf{X} + \mathbf{b}) . \quad (2.5.2.5)$$

This can of course be generalized to the case where both the input \mathbf{X} and consequently the weights $\boldsymbol{\omega}$ and biases \mathbf{b} are multi-dimensional tensors. In the following, we will restrict the discussion to the case presented in Eq. (2.5.2.5) for the sake of simplicity and because this covers the cases used in this thesis, *cf.* Ch. 3.

Hence, the so-called *layer* defined in Eq. (2.5.2.5), *viz.* the usage of multiple neurons to process signals from incoming neurons to a multi-dimensional output, can be summarized as a nonlinear activation function applied component-wise to the sum of a matrix-vector-multiplication and a bias vector, where the weight matrix $\boldsymbol{\omega}$ and the bias vector \mathbf{b} are trainable parameters of each layer separately. The activation function a on the other hand is a hyperparameter chosen arbitrarily but fixed before the training process described in Sec. 2.5.2.3. This fundamental building block of neural networks, also called a *dense layer* or *fully-connected layer* due to the fact that every input neuron is connected to every output neuron, is therefore equivalent to a *linear regression with a nonlinear activation function*. On the right hand side of Fig. 2.7 the calculations done in Eq. (2.5.2.5) are depicted diagrammatically with the biological counterpart on the left hand side for illustration of the analogy between both.

Besides allowing for the application of a single linear regression to a nonlinear problem, the activation function also facilitates stacking multiple layers by using the output of one layer as the input to the next and therefore increasing the number of functions the neural network is able to approximate. The layers inbetween the *input layer* and the *output layer* are called *hidden layers*. Let \mathbf{X} denote the input to a network consisting of two stacked layers, where the variables in the i -th layer are denoted using a superscript (i), then the output $\mathbf{O}(\mathbf{X})$ of this *2-layer neural network* reads

$$\mathbf{O}(\mathbf{X}) = \mathbf{z}^{(2)}(\mathbf{z}^{(1)}(\mathbf{X})) \quad (2.5.2.6)$$

$$= \mathbf{z}^{(2)}(a^{(1)}(\boldsymbol{\omega}^{(1)} \cdot \mathbf{X} + \mathbf{b}^{(1)})) \quad (2.5.2.7)$$

$$= a^{(2)}(\boldsymbol{\omega}^{(2)} \cdot a^{(1)}(\boldsymbol{\omega}^{(1)} \cdot \mathbf{X} + \mathbf{b}^{(1)}) + \mathbf{b}^{(2)}) \quad (2.5.2.8)$$

The number of neurons in each layer, also known as the *size* of the corresponding layer, defines the dimensions of the weight matrix and the bias vector and

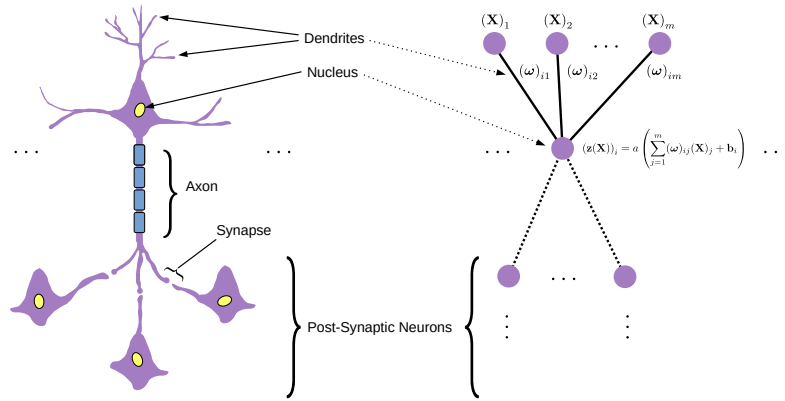


Figure 2.7: Schematic of a biological neuron (left side) and the correspondence to artificial neurons (right side), with the similarities indicated via dotted arrows. The dendrites correspond to the connections between incoming information \mathbf{X} and the current neuron i , with the strength of the dendrites modeled by the weights $(\omega)_{ij}$. The incoming signals are then gathered and processed in the nucleus or in the calculation of the activation $(z(\mathbf{X}))_i$ respectively. The output from neuron i is then passed to the post-synaptic neurons, *i.e.* its successors, which in the case of artificial neural networks would be the subsequent hidden or output layer.

can of course be varied. Furthermore, it is a hyperparameter, which is chosen before training the weights of the neural network and stays constant throughout the training procedure.

If a linear function would be chosen as activation, then the introduction of additional layers does not change the class of functions the neural network can represent, as a composition of linear functions is still a linear function. For example, if in Eq. (2.5.2.8) one chose $a^{(1)}$ to be the identity, $a^{(1)}(x) = x$, then Eq. (2.5.2.8) could be written as

$$\mathbf{O}(\mathbf{X}) = a^{(2)}(\boldsymbol{\omega}^{(2)} \cdot a^{(1)}(\boldsymbol{\omega}^{(1)} \cdot \mathbf{X} + \mathbf{b}^{(1)}) + \mathbf{b}^{(2)}) \quad (2.5.2.9)$$

$$= a^{(2)}(\boldsymbol{\omega}^{(2)} \cdot (\boldsymbol{\omega}^{(1)} \cdot \mathbf{X} + \mathbf{b}^{(1)}) + \mathbf{b}^{(2)}) \quad (2.5.2.10)$$

$$= a^{(2)}(\boldsymbol{\omega}^{(2)} \cdot (\boldsymbol{\omega}^{(1)} \cdot \mathbf{X}) + \boldsymbol{\omega}^{(2)} \cdot \mathbf{b}^{(1)} + \mathbf{b}^{(2)}) \quad (2.5.2.11)$$

$$= a^{(2)}((\boldsymbol{\omega}^{(2)} \cdot \boldsymbol{\omega}^{(1)}) \cdot \mathbf{X} + (\boldsymbol{\omega}^{(2)} \cdot \mathbf{b}^{(1)} + \mathbf{b}^{(2)})) \quad (2.5.2.12)$$

$$= \bar{a}^{(1)}(\bar{\boldsymbol{\omega}}^{(1)} \cdot \mathbf{X} + \bar{\mathbf{b}}^{(1)}) \quad (2.5.2.13)$$

such that the same function implemented by the 2-layer neural network with

linear activation $a^{(1)}$ could be represented by a 1-layer neural network with

$$\bar{a}^{(1)} = a^{(2)} \quad (2.5.2.14)$$

$$\bar{\omega}^{(1)} = \omega^{(2)} \cdot \omega^{(1)} \quad (2.5.2.15)$$

$$\bar{\mathbf{b}}^{(1)} = \omega^{(2)} \cdot \mathbf{b}^{(1)} + \mathbf{b}^{(2)}. \quad (2.5.2.16)$$

Hence, in order to increase the range of functions with an increase in the number of layers, the corresponding activation functions have to be nonlinear. If the neural network consists of a large number of these non-linearly activated layers, it is called *deep*, lending its name to the field of *deep learning*.

While the fully-connected layer as defined in Eq. (2.5.2.5) is the fundamental building block of most neural networks, there is of course an abundance of similar building blocks, like *long short-term memory (LSTM) layers*⁶¹ or *self-attention layers*⁶² which exceed the scope of this thesis. In the following subsection we will introduce another concept, that was used in this thesis, namely the convolutional layer.

2.5.2.2 Convolutional Layers

As training neural networks is akin to fitting a very high dimensional function to the available set of data, it is prone to *overfitting* the data, *i.e.* the network learns to perfectly reproduce the data available during training but fails to generalize to previously unseen examples, *cf.* Sec. 2.5.3. Furthermore, the sheer number of parameters makes both the evaluation of the neural network given an input as well as its training very time consuming, as the space of possible parametrizations becomes huge. While the training process will be discussed in more details in Sec. 2.5.2.3, it is obvious from the argument above, that reducing the number of trainable parameters is one way to alleviate this problem. For data points with spatial correlations, *e.g.* for images, where pixels close to each other are correlated and a permutation of pixels would destroy the image's interpretability, there is a common technique called *convolutional layers*.

In the following, we will restrict this discussion to the one dimensional case, *i.e.* the input to the neural network is a m -dimensional vector $\mathbf{X} \in \mathbb{R}^m$ where dimensions close to each other are correlated in general. Here, we will only cover this special case as the convolutional layers used in this thesis fall into this category exclusively, *cf.* Ch. 3. The spatial correlations in the applications discussed in Ch. 3 stem from the fact, that the data presented to the neural network are discretized representations of radial distribution functions and

pair potentials, which are spatially correlated for obvious reasons.

Instead of connecting every input neuron to every output neuron with weights for every single connection, as is done in the fully-connected layer, the convolutional layer comprises of a so-called *kernel* of trainable weights, which is applied in a convolutional manner multiple times to different parts of the input. This approach is inspired by the receptive field in the visual cortex, where incoming photons are not processed by all neurons but only by neurons which are close to the visual stimulus. The operation of the i -th neuron in a convolutional layer for the one dimensional case then reads

$$(\mathbf{z}(\mathbf{X}))_i = \sum_{j=1}^k (\mathbf{X})_{i+j-1-\lfloor \frac{k}{2} \rfloor} (\mathcal{K})_j + (\mathbf{b})_i \quad (2.5.2.17)$$

where the dimensionality of the trainable kernel \mathcal{K} , also called the *kernel size* $k = \dim(\mathcal{K})$, is a hyperparameter that is chosen before the training procedure. As becomes clear from Eq. (2.5.2.17), the kernel is the same for all output neurons, which means the learned weights contained in it are shared. This reduces the number of learnable parameters drastically and, by virtue of the sum only covering surrounding entries of the input vector, introduces a local dependence of the output neurons on a subset of the input neurons rather than a global dependence. Actually, the operation described above is not a *convolution* but rather a *cross-correlation*, which is why the term “convolutional layer” is in fact a misnomer. However, both of these mathematical operations are related via

$$(f(t) \star g(t))(\tau) = \int_{-\infty}^{\infty} f^*(t)g(\tau + t)dt \quad (2.5.2.18)$$

$$= - \int_{\infty}^{-\infty} f^*(t)g(\tau + t)dt \quad (2.5.2.19)$$

$$= \int_{\infty}^{-\infty} f^*(-t)g(\tau - t)dt \quad (2.5.2.20)$$

$$= (f^*(-t) * g(t))(\tau) \quad (2.5.2.21)$$

where f and g are two functions. The symbols \star and $*$ denote the cross-correlation and the convolution respectively. In the discretized case and for real numbers as in Eq. (2.5.2.17), Eq. (2.5.2.21) implies that a proper convolution is obtained by flipping the kernel before performing the cross-correlation in Eq. (2.5.2.17). As the parameters in the kernel of the convolutional layer are a result of the training process anyways, this does not have any effect as long as one follows a consistent convention. In the following, we will keep the standard convention

of using Eq. (2.5.2.17) and calling the corresponding operation a “convolution”.

There are different ways of dealing with the edge cases of the sum in Eq. (2.5.2.17), where the corresponding index of the component of \mathbf{X} is not in the range of $[1, \dim(\mathbf{X})]$. In the so-called *valid mode*, one chooses to ignore the indices i for which this is the case, resulting in a smaller number of output neurons. In the *same mode*, the vector \mathbf{X} is first padded with values, typically but not necessarily zero, filling the “missing” indices, before applying Eq. (2.5.2.17). Finally, in the *circular mode*, the index of \mathbf{X} in Eq. (2.5.2.17) is subject to a modulo-operation so that the operation is performed on a ring and becomes akin to periodic boundary conditions.

2.5.2.3 Backpropagation in Neural Networks

While there exist methods to train neural networks without the calculations of gradients, so-called *derivative-free optimization algorithms*, like simulated annealing or genetic algorithms, state-of-the-art models are almost exclusively trained by *gradient-based methods*, as they are found to perform better for larger problems.⁶³ Gradient-based methods are iterative procedures, which consist of alternately calculating updates of the weights of the neural network and adjusting the weights accordingly. The most common gradient-based algorithm for training neural networks is *backpropagation*, an algorithm devised by Stuart Dreyfuss in 1973⁶⁴ and then adapted to neural networks by Paul Werbos in his dissertation⁶⁵ as well as by Rumelhart, Hinton and Williams.⁶⁶

In backpropagation, the changes of the weights of the neural network are calculated using the gradient of a *loss function* \mathcal{L} , which measures the performance of the neural network given a set of examples. This loss function can but need not be equal to the performance measure P described in Sec. 2.5.1, yet a low value of the loss function should ideally imply a better performance with regards to P . An example for typical deviations of the loss function from P are regularizing terms, which will be discussed in Sec. 2.5.3.

In principle, the calculations of gradients can be done straight-forward using the chain rule of differentiation. In practice however, this is computationally not feasible for neural networks comprised of many large layers, as the error calculation includes computing all derivatives inbetween the corresponding layer and the output, which becomes intractable quite fast. Backpropagation in principle does apply the chain rule as well, but it reuses the error of the previous layer when doing so, making it an example of *dynamic programming*. Besides this dynamic programming approach the usage of *automatic differentiation* or

autodiff facilitates these gradient-based optimization schemes, as they allow to calculate the derivative of a function accurate to numerical precision without having to use any symbolic or numerical differentiation.

While the backpropagation algorithm provides a rule for how to calculate the gradient of the loss function with respect to the weights of the neural network given an example, there is some freedom in the choice of how to proceed with a set of examples and how to update the weights. Here, we will only shortly mention some popular choices and refer to the relevant literature for more information, *e.g.* to Ref. [57].

For the number of training examples in each update step, one might choose to calculate the average of the gradients over all available training examples as is done in *batch gradient descent*. This approach yields stable updates and therefore a more robust training procedure, but it is also very inefficient as for every single update, all training examples have to be fed through the network. In *stochastic gradient descent* the update is applied after every training example, which results in very frequent updates yet tends to be rather unstable, as it is subject to a lot of noise in the updates due to the small sample size and hence strongly fluctuating gradients. The *mini-batch gradient descent* methods lies in between these extremes by calculating the gradients on a subset of all training examples, with the corresponding *batch size* k being a hyperparameter of the training procedure, and can therefore be adjusted to mitigate the disadvantages of the other methods. By choosing the batch size as $k = 1$ one recovers stochastic gradient descent and by setting it to the number of all available training examples one recovers batch gradient descent.

In the aforementioned batch gradient descent methods, the update rule for a weight $\omega_{ij}^{(l)}$ of a neural network \mathbf{O} given a set of examples $E_k = \{\mathbf{X}_i, \mathbf{Y}_i \in E | i \in \{1, \dots, k\}\}$ reads

$$\omega_{ij}^{(l)} \leftarrow \omega_{ij}^{(l)} - \alpha \frac{1}{k} \sum_{(\mathbf{X}_h, \mathbf{Y}_h) \in E_k} \frac{\partial \mathcal{L}(\mathbf{O}(\mathbf{X}_h), \mathbf{Y}_h)}{\partial \omega_{ij}^{(l)}} \quad (2.5.2.22)$$

with α denoting the *learning rate*, *i.e.* a scaling factor for the update.

There exist multiple variations of this update rule, like choosing examples with or without replacement or not weighting the gradients from different examples uniformly. Besides that, there exist more sophisticated update algorithms like ADAM,⁶⁷ which exceed the scope of this thesis.

2.5.3 Bias-Variance Tradeoff and Regularization

In machine learning, one mainly distinguishes two types of errors, namely *bias* and *variance*.

The bias refers to the error that stems from wrong assumptions of the ML algorithm, *i.e.* its inability of fully finding the underlying correlations between input and output. Hence, if a model has a high bias, it means it is not able to accurately predict the target even on previously seen examples, which is referred to as *underfitting*, as the model does not fit the target mapping properly. Typically, a model suffers from high bias if the target mapping is not part of the class of functions covered by the chosen model architecture or if its complexity is not sufficient to represent the target mapping. This is why often the bias can be reduced by either choosing a different model architecture or by increasing the model's complexity, *e.g.* by introducing more or larger layers in the case of neural networks. The name “bias” comes from the fact, that the model has an overall tendency of falsely or inaccurately predicting the target, regardless of the source of the data.

The variance on the other hand describes the *generalization error* of the model, *i.e.* the error caused by the model being trained to reproduce the training examples so closely, that a small change in input examples might lead to vastly different results. This is called *overfitting*, as the model is typically able to reproduce the target mapping very closely on training examples but is not able to return accurate predictions on previously unseen examples, in particular examples which slightly lie outside the training distribution. This often occurs, when the model complexity is too large so that it can distinguish training examples by tiny differences, which is why a high variance can often be reduced by decreasing the class of functions representable by the model, *i.e.* by reducing its complexity. Furthermore, one might introduce additional constraints to the training object to avoid overfitting the training data, as will be discussed in this section. The term “variance” refers to the model's predictions varying significantly with only slight changes in the input.

While it is fairly easy to detect a high bias by analyzing the performance of a model on the training set and comparing it to the performance requirements or baselines like *human level performance*, detecting a high variance typically requires the usage of a validation data set, whose purpose is to determine the model's capabilities of generalizing to data, that is not used in the tuning of the model parameters, *cf.* Sec. 2.5.1.2.

As mentioned before, a high bias can typically be reduced by either choosing a different type of ML algorithm or architecture or by increasing the corresponding model complexity. A high variance on the other hand can often be reduced by reducing the model's complexity in order to limit the classes of functions, which can be represented by these models. As these mitigation strategies are opposing, *i.e.* a reduction of the bias often leads to an increase in variance and *vice versa*, one often speaks of the *bias-variance tradeoff*.

Even though there exists a multitude of alternative approaches to reduce said errors, the above tradeoff is still encountered in almost all cases. However, they might differ in the effect they have on these errors and hence their extent and severity of this tradeoff. One of the approaches that has proven to be fruitful is the usage of *regularization* terms, in which additional terms are incorporated into the loss functions, which are supposed to reduce range of values of the the model's weights to in turn limit the model's capabilities of overfitting. Typical regularization methods, which were also used in Ch. 3, are the L_1 - and L_2 -regularization, which read

$$L_1(\Theta) = \sum_{l,i,j} |\omega_{ij}^{(l)}| + \sum_{l,i} |\mathbf{b}_i^{(l)}| \quad (2.5.3.1)$$

$$L_2(\Theta) = \sum_{l,i,j} |\omega_{ij}^{(l)}|^2 + \sum_{l,i} |\mathbf{b}_i^{(l)}|^2 \quad (2.5.3.2)$$

where $\omega_{ij}^{(l)}$ are the model's weights and $\mathbf{b}_i^{(l)}$ its biases. By supplementing this loss term to the original loss function weighted by a hyperparameter, the model has to optimize the previous objective while additionally limiting the the size of its weights and biases, such that Eqs. (2.5.3.1) and (2.5.3.2) serve as soft *Lagrange multipliers*. By limiting the size of the weights and biases, the model cannot focus too strongly on single features but has to accurately predict the target using different, less strongly weighted features which encourages abstraction.

Another tried and tested method for reducing variance also used in this thesis is *dropout regularization* or simply *dropout*. Dropout regularization means that during training some of the weights of the neural network will be multiplied by zero randomly, *i.e.* the corresponding inputs to the neuron are effectively discarded. This forces the model to again not focus too much on single weights but distribute the information to multiple weights in the corresponding layer, as the model is regularly trained with these heavily-focused-on weights deactivated, such that other weights have to still be able to propagate the relevant information. The effect of this is again an encouragement of abstraction and therefore better generalization capabilities.

Besides these methods there is a plethora of other effective techniques reducing overfitting, which exceed the scope of this thesis and can be found in the relevant literature, such as Ref. [57].

CHAPTER 3 --- BoltzmaNN: Predicting effective pair potentials and equations of state using neural networks

The following chapter has been published in word as:⁶⁸

Fabian Berressem and Arash Nikoubashman, "BoltzmaNN: Predicting effective pair potentials and equations of state using neural networks", *The Journal of Chemical Physics* **154**, 124123 (2021) <https://doi.org/10.1063/5.0045441>

Using the CRediT (Contributor Roles Taxonomy) author statement,⁶⁹ the contributions to this work are as follows:

Fabian Berressem: Methodology (lead); Conceptualization (equal); Software (lead); Validation (lead); Formal analysis (lead); Investigation (equal); Data curation (equal); Writing - original draft (lead); Writing - review & editing (equal); Visualization (lead)

Arash Nikoubashman: Conceptualization (equal); Methodology (supporting); Software (supporting); Validation (supporting); Formal analysis (supporting); Investigation (equal); Resources (lead); Data curation (equal); Writing - original draft (supporting); Writing - review & editing (equal); Supervision (lead); Project administration (lead); Funding acquisition (lead)

Abstract

Neural networks (NNs) are employed to predict equations of state from a given isotropic pair potential using the virial expansion of the pressure. The NNs are trained with data from molecular dynamics simulations of monoatomic gases and liquids, sampled in the NVT ensemble at various densities. We find that the NNs provide much more accurate results compared to the analytic low-density limit estimate of the second virial coefficient and the Carnahan-Starling equation of state for hard sphere liquids. Further, we design and train NNs for computing (effective) pair potentials from radial pair distribution functions, $g(r)$, a task which is often performed for inverse design and coarse-graining. Providing the NNs with additional information on the forces greatly improves the accuracy of the predictions, since more correlations are taken into account; the predicted potentials become smoother, are significantly closer to the target potentials, and are more transferable as a result.

3.1 Introduction

Understanding and predicting the relationship between the (macroscopic) properties of a material and its (microscopic) building blocks is one of the key challenges in materials research and physics. One important goal in statistical physics is the accurate prediction of the phase behavior on the basis of the (effective) pair potential $U(r)$ acting between the particles. According to van der Waals' theorem of corresponding states, all simple fluids obey the same reduced equation of state (EOS), if the thermodynamic variables are rescaled by their value at the critical point. However, this law only applies for systems with conformal pair potentials, *i.e.*, when the potentials can be fully superimposed by adjusting the interaction strength and particle diameter which is rarely the case in practice. Noro and Frenkel extended this principle by including the reduced second virial coefficient for quantifying the effective range of the attraction.³⁹ This extended approach can provide accurate predictions for pair potentials which are characterized by attractive interactions with ranges much smaller than the particle size,^{39,70} but it is expected to fail for more complex pair potentials which, *e.g.*, include a repulsive barrier. Thus, alternative prediction tools are highly desirable, especially given that a large number of (effective) pair potentials in soft matter are bounded or have repulsive barriers.^{71–75}

Progress in this field has wide implications, not just in terms of our fundamental understanding, but also due to the large number of potential technological applications. Various mechanical, optical, and electronic material properties

critically depend on the degree of ordering of their atomic or (macro)molecular constituents. In the conventional *forward design* approach, the development of new materials typically begins with designing candidate building blocks that are expected to lead to the desired properties. Then these candidates are created, tested, and, if necessary, modified, until the compound with the wanted properties has been identified. However, such iterative optimization procedures are often rather time- and resource consuming. Therefore, good initial candidates are required to achieve convergence in a reasonable time frame. Due to these inherent issues, there has been a recent paradigm shift towards *inverse design* processes, where the building blocks are inferred from the desired target properties. This pathway has been explored for a range of soft materials,^{76,77} including athermal granular media,⁷⁸ colloids,^{79,80} and block copolymers.^{81,82} In recent years, informatics-driven approaches have gained popularity that utilize machine learning algorithms on large databases to identify previously unrecognized patterns and to predict new candidate materials.^{83–90}

This inverse design process is strongly related to the task of top-down coarse-graining, where the goal is to derive effective (pair) potentials for a system with a reduced number of degrees of freedom, while preserving selected target properties of the original fine-grained representation. Several methods have been developed for this task, including Reverse Monte Carlo (RMC),^{46,91,92} Iterative Boltzmann Inversion (IBI),⁹³ and simulated annealing-based optimization.⁹⁴ These techniques have been employed successfully for, *e.g.*, developing effective pairwise potentials from experimental structure measurements^{95–97} and for coarse-graining atomistic simulations.^{93,98–100} Achieving transferability and representability of such coarse-grained models is, however, a key challenge,^{101,102} given that the multi-body potential of mean force is usually approximated by an effective pair potential.^{36,46,91–94,101,102} Further, although the IBI scheme should in principle provide a unique solution for a given radial pair distribution function $g(r)$,⁴⁵ convergence of this iterative procedure is not guaranteed in practice.¹⁰² Various strategies have been devised for improving these and other aspects of coarse-graining, such as the addition of thermodynamic constraints,^{93,103,104} and the development of improved methods is an active field of research.^{105–109}

In this work, we employ artificial neural networks (NNs) for predicting the EOS from a given isotropic pair potential $U(r)$ using the virial expansion of the pressure. Further, we design and train NNs for computing (effective) pair potentials from a given $g(r)$. The training and test data for the NNs are generated from Molecular Dynamics (MD) simulations of monoatomic gases and liquids in the canonical ensemble. The NNs developed in this work as well as example scripts are available online.¹¹⁰ The rest of this manuscript is organized as follows. In Sec. 3.2, we provide a brief summary of the numerical methods we used

as well as how the data were generated. In Sec. 3.3, we present our results, where Sec. 3.3.1 focuses on using NNs for determining the EOS from a given pair potential, and Sec. 3.3.2 discusses how NNs can be used for computing (effective) pair potentials from the radial pair distribution function. In Sec. 3.3.3 we present typical use case scenarios for our approach. Section 3.4 contains our conclusions and a brief outlook

3.2 Methodology

3.2.1 Design of Neural Networks

We used three different NN architectures in this work, *i.e.*, fully connected dense NNs (DN), convolutional NNs (CN), and U-Nets¹¹¹ (UN). The DNs consist of layers of neurons, where all neurons between two subsequent layers are connected to each other. Operations in DNs are limited to simple matrix-matrix and matrix-vector multiplications, combined with (nonlinear) activation functions to break linearity. The CNs use convolutional operations which, compared to a DN, drastically reduce the number of parameters necessary by considering only local correlations between points. Therefore, CNs are useful for strongly correlated data, as is typically the case in image processing but also in our problem. In UNs, the input is first processed using convolutional layers then upsampled again and concatenated to a former stage of the processing, hence extracting features and combining them with the original input for further processing.

The accuracy of an NN depends strongly on the information that is provided to it, *e.g.*, the average particle number density and/or the force. Also, the format in which this information is represented plays an important role. For instance, one could use $\exp[-\beta U(r)]$ rather than $\beta U(r)$ [with inverse thermal energy $\beta \equiv 1/(k_B T)$], which may be favorable for pair potentials with a strong repulsion combined with an attraction of significantly smaller strength. A detailed study on the effect of these parameters is provided in Sec. 3.3.

Another key aspect of the NNs is the loss function, which essentially controls the properties that should be optimized by the NN. The loss functions used in this work were combinations of the squared error (SE), the absolute error (AE), and the logcosh error (LE). For a given pair of target and prediction, (y, \hat{y}) , the basic losses are

$$l_{\text{SE}}(y, \hat{y}) = (y - \hat{y})^2, \quad (3.2.1.1)$$

$$l_{\text{AE}}(y, \hat{y}) = |y - \hat{y}|, \quad (3.2.1.2)$$

$$l_{\text{LE}}(y, \hat{y}) = \ln [\cosh (y - \hat{y})]. \quad (3.2.1.3)$$

The loss of the entire output vector is then determined as the mean over the losses of the individual nodes. For instance, the mean squared error (MSE) is given by

$$L_{\text{MSE}}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{d} \sum_{i=1}^d l_{\text{SE}}(y_i, \hat{y}_i), \quad (3.2.1.4)$$

with $\mathbf{y}, \hat{\mathbf{y}} \in \mathbb{R}^d$ and dimensionality d of the vectors (e.g. $d = 50$ for predicting pair potentials). Analogous expressions are used for L_{MAE} and L_{MLE} . We indicate the average loss taken over multiple output vectors through angular brackets, $\langle \dots \rangle$.

Expressions like Eq. (3.2.1.4) measure the difference between two single points of \mathbf{y} and $\hat{\mathbf{y}}$, but do not capture the correlations between neighboring points. To ensure that the predicted potentials vary smoothly with r , it is therefore helpful to include a Laplace-like term as well as loss terms correlating the differences of the potential at different distances

$$L_U = L_{\text{MLE}} + \alpha_{\Delta} L_{\Delta} + \sum_{k=1}^4 L_k \alpha^k \quad (3.2.1.5)$$

with weight $\alpha_{\Delta} = 2$. The terms L_k can optionally be multiplied by a factor $\alpha \leq 1$ to reduce their contribution with increasing distance ($\alpha = 1$ was used in this work). The term L_{Δ} is the discretized Laplace term, which in the case of l_{SE} reads

$$L_{\Delta} = \frac{1}{d-2} \sum_{i=1}^{d-2} l_{\text{SE}}(y_{i+2} - 2y_{i+1} + y_i, \hat{y}_{i+2} - 2\hat{y}_{i+1} + \hat{y}_i). \quad (3.2.1.6)$$

The loss terms L_k in Eq. (3.2.1.5) are given by the loss function applied to the difference of potential values being k indices apart. For example, with l_{SE} this term reads

$$L_k = \frac{1}{d-k} \sum_{i=1}^{d-k} l_{\text{SE}}(y_{i+k} - y_i, \hat{y}_{i+k} - \hat{y}_i). \quad (3.2.1.7)$$

To see how this definition naturally introduces spatial correlations, one can rewrite L_k from Eq. (3.2.1.7) as follows

$$L_k = \frac{1}{d-k} \sum_{i=1}^{d-k} [l_{\text{SE}}(y_{i+k}, \hat{y}_{i+k}) + l_{\text{SE}}(y_i, \hat{y}_i) - 2(y_{i+k} - \hat{y}_{i+k})(y_i - \hat{y}_i)]. \quad (3.2.1.8)$$

In this representation of L_k , it is clear that the last term in Eq. (3.2.1.8) introduces correlations between values at different distances.

All NNs have been constructed and trained using Tensorflow v. 1.13.1.¹¹² The networks were trained using an Adam (adaptive moment estimation) optimizer⁶⁷ with a learning rate of 0.001. In order to find a suitable NN topology, an extensive grid search was conducted for the three network architectures, where we systematically studied the influence of the loss function as well as the number and shape of the layers, *i.e.*, the width or number of filters and kernel size. During this grid search, the NNs were trained for 10^4 epochs without any optional parameters (see Sec. 3.3 below). For each architecture, we chose the NN with the lowest $\langle L_{\text{MSE}} \rangle$ and $\langle L_{\text{MAE}} \rangle$ evaluated through 4-fold cross-validation, and trained these NNs then for 2×10^5 epochs with optional parameters. For the final benchmarks, the selected NNs were trained for 5×10^5 epochs using all data from the training and validation sets.

3.2.2 Generation of data

In order to generate training and test data for the NNs, we performed a series of MD simulations of monoatomic gases and liquids in the canonical ensemble, and determined the resulting pressures P and radial pair distribution functions $g(r)$. Model parameters and physical quantities are expressed in fundamental units of σ , ε , m , and $\tau = \sqrt{m\sigma^2/\varepsilon}$ for length, energy, mass, and time, respectively. Pair potentials $U(r)$ were generated as spline functions with cutoff distances $r_{\text{cut}} \in (0, 5\sigma]$. The number of base points was chosen randomly in the range $n \in [6, 10]$, and the base points were distributed randomly at distances in the range $[0, r_{\text{cut}}]$ with magnitudes in the interval $[0, 15\varepsilon]$. Then a smoothing spline function was applied to connect these points. The degree of the spline function was randomly drawn between $k = 2$ and $k = 5$ to allow for a large variety of shapes [note that $k \geq 2$ is required to guarantee differentiability of $U(r)$]. To ensure physically meaningful pair potentials, the splines were fitted with the constraint $F(r_{\text{cut}}) = -\frac{\partial}{\partial r}U(r)|_{r_{\text{cut}}} = 0$. Further, the potentials were shifted so that $U(r_{\text{cut}}) = 0$. Hard-core interactions were included in selected pair potentials by adding the Weeks-Chandler-Andersen (WCA) potential¹⁰ with diameter σ and interaction strength ε .

Using this procedure, 657 potentials were generated, with 344 of them having an additional hard-core contribution. For the MD simulations, tabulated potentials were created with 200 evenly distributed points in the range $[0, r_{\text{cut}}]$. For the optimization of the NNs, these tables were then further downsampled to 50 points, evenly distributed in the interval $[0, 5\sigma]$. Figure 3.1(a) shows a selection of potentials which were generated using our procedure, while the

resulting radial pair distribution functions are plotted in Fig. 3.1(b,c).

The MD simulations were conducted in a cubic simulation box with edge length $L = 40 \sigma$ and periodic boundary conditions in all directions. The temperature was fixed at $k_B T = \varepsilon$ using a Langevin thermostat, and the equations of motions were solved using the standard Velocity-Verlet algorithm with a time step of $\Delta t = 0.005 \tau$. Simulations were conducted at four different particle number densities, *i.e.*, $\rho = 0.125 \sigma^{-3}$ ($N = 8000$), $\rho = 0.244 \sigma^{-3}$ ($N = 15625$), $\rho = 0.512 \sigma^{-3}$ ($N = 32768$), and $\rho = 1.0 \sigma^{-3}$ ($N = 64000$). Each simulation was first run for 5×10^6 MD steps for equilibration, and then for additional 5×10^6 MD steps to sample P and $g(r)$ [see Fig. 3.1(b,c) for examples]. For the discretization of the calculated $g(r)$ we chose 200 points that were evenly distributed in the interval $[0, L/2]$. In total, 1678 simulations were conducted, with 529 containing potentials with hard-core repulsion. Simulations were discarded from our analysis, which did not reach equilibrium in the allotted time or formed heterogeneous structures (see ESI for more information), leading to a final number of 891 valid simulations. Of those, 790 simulations were used as training and validation sets, while the remaining 101 were reserved for testing only. All simulations were performed on GPUs using the HOOMD-blue software package (v. 2.4.2).¹¹³

3.3 Results

3.3.1 Predicting the equation of state

We applied NNs for predicting the pressure P at a given particle number density ρ and temperature T from the employed pair potential $U(r)$ *via* three different strategies:

1. The pair potential is directly mapped to the pressure, $\beta U(r), \rho \mapsto \beta P$;
2. The pair potential is first mapped to an effective, density-dependent second virial coefficient $\beta U(r), \rho \mapsto B_2^*$ and then to the pressure, $\beta P = \rho + \rho^2 B_2^*$;
3. The pair potential is first mapped to a set of n virial coefficients $\beta U(r) \mapsto \mathcal{B}_n := \{B_i \mid 2 \leq i \leq 5\}$ and then to the pressure, $\beta P = \rho + \sum_{B_i \in \mathcal{B}_n} B_i \rho^i$.

Figure 3.2 shows schematic representations of the NNs used for predicting βP from a given pair potential $\beta U(r)$ and density ρ . For the strategies $\beta U(r), \rho \mapsto \beta P$ and $\beta U(r), \rho \mapsto B_2^*$, the density was included as an additional input after convolutional operations, so that the density-dependence is learned

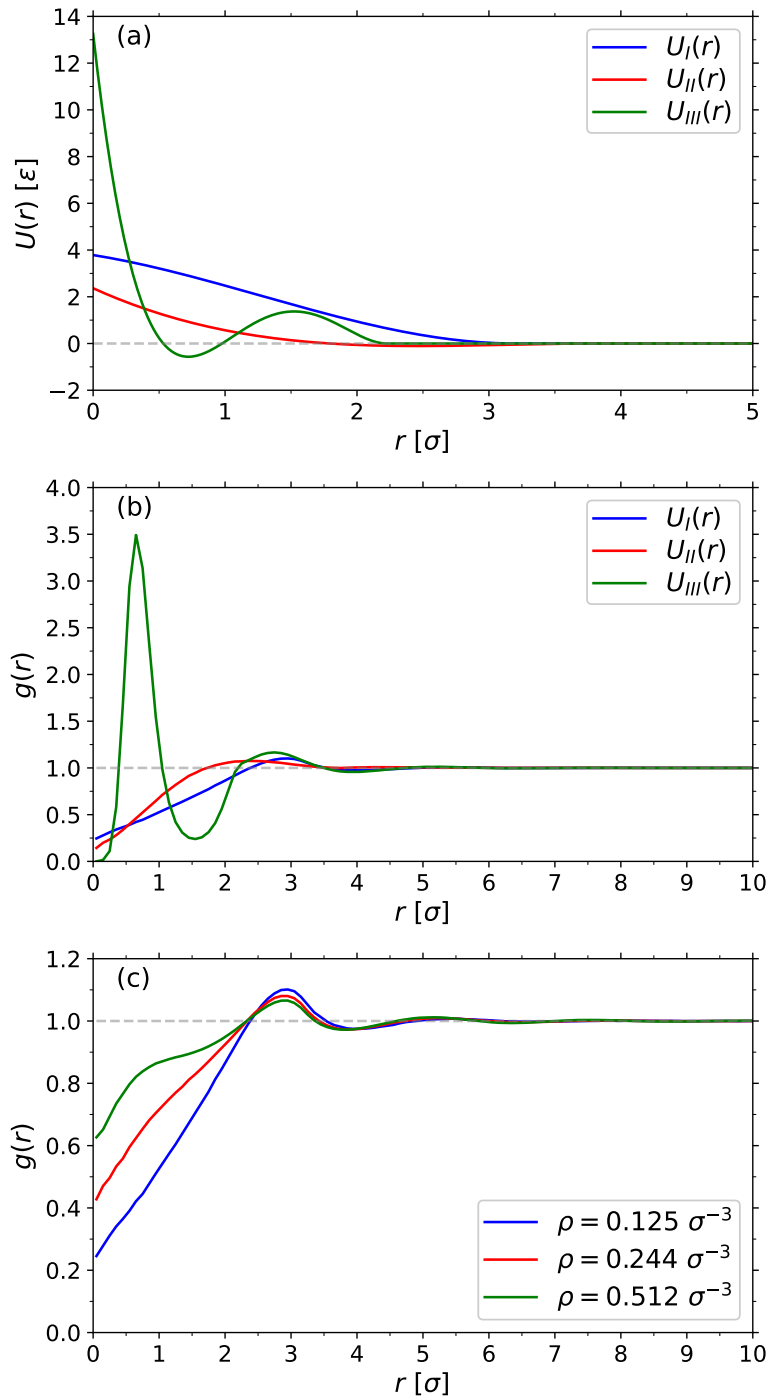


Figure 3.1: (a) Three selected pair potentials $U(r)$ used in this work, and (b) corresponding radial pair distribution functions, $g(r)$, recorded at particle number density $\rho = 0.125 \sigma^{-3}$. (c) $g(r)$ for U_I at three investigated values of ρ , as indicated.

by the NN. The calculation of βP for the network calculating \mathcal{B}_n is achieved *via* layers calculating the scalar product or sum as indicated in the schematic. These layers do not contain any adjustable parameters, so that the learned virial coefficients \mathcal{B}_n are density independent, while still allowing for the loss function to be applied to the density-dependent pressure βP .

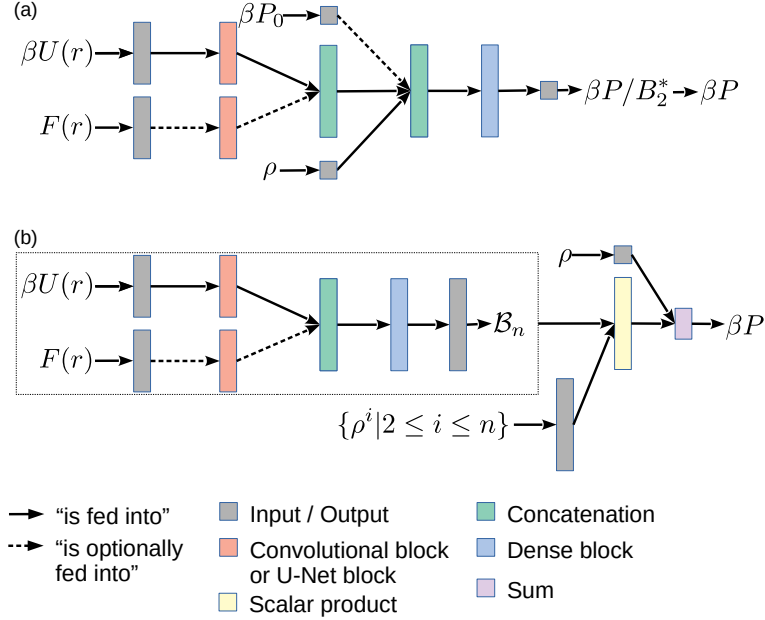


Figure 3.2: Schematic representation of the NNs used for computing the pressure βP (a) directly or *via* the effective second virial coefficient, B_2^* , and (b) through a set of $n = 5$ density-independent virial coefficients $\mathcal{B}_n = \{B_i\}$. The dotted box in (b) indicates the central part of the NN, while the operations outside of this box do not contain learnable parameters and are only used to calculate βP .

The only required input parameters of our NNs are $\beta U(r)$ and ρ . The temperature dependence of the pressure is included through the prefactor β in the input and the output ($\beta = 1/\varepsilon$ by default). In principle, one could directly use the tabulated pair potentials $\beta U(r)$ as the input of the NNs. In practice, however, this might cause numerical issues for strongly repulsive potentials, since a further increase of the interaction strength beyond a certain threshold has little or no effect. Therefore, it is sensible to reduce the search space and minimize the risk of overfitting by limiting the maximum repulsion of $\beta U(r)$ during the optimization of the NNs. The range of the pair potential $\beta U(r)$ was optionally constrained either by representing it in exponential space, *i.e.*, $\exp[-\beta U(r)]$, or by using a clipped potential, $\beta \bar{U}(r) = \max[\beta U(r), \beta U_{\text{cut}}] \forall r$ with $\beta U_{\text{cut}} = 20$. The choice of U_{cut} is somewhat arbitrary but our tests indicated that the specific

value does not significantly alter the outcome as long as U_{cut} separates soft and hard potentials.

Furthermore, we tested how additional information can improve the prediction accuracy. The force $F(r) = -\frac{\partial}{\partial r}\beta U(r)$ was optionally included in the NNs, by applying the same operations as for $\beta U(r)$ and then concatenating the outputs for further processing in fully dense layers (see Fig. 3.2). If $\beta \bar{U}(r)$ was used as the input, then the force was calculated before clipping to avoid unphysical discontinuities. If the input was $\exp[-\beta U(r)]$, then $-\frac{\partial}{\partial r}\exp[-\beta U(r)]$ was used as the force input. Note that including the derivative almost doubles the number of nodes in the network. For the NNs learning B_2^* or βP , we optionally provided also a pressure estimate βP_0 , calculated using the analytic second virial coefficient B_2 . In a homogeneous system with isotropic interactions, B_2 is readily available from $U(r)$ ¹¹⁴

$$B_2(T) \approx -2\pi \int_0^\infty f(r, T) r^2 dr, \quad (3.3.1.1)$$

with Mayer f -function

$$f(r, T) = \exp[-\beta U(r)] - 1. \quad (3.3.1.2)$$

Thus for $\rho \rightarrow 0$, the pressure βP_0 can be directly computed from the pair potential $U(r)$

$$\beta P_0 = \rho - 2\pi\rho^2 \int_0^\infty f(r, T) r^2 dr. \quad (3.3.1.3)$$

In what follows, we will use the following naming convention for the NNs: For a given architecture X and given options y , we name the network X/y , *e.g.*, for a UN with force information we use UN/f. The abbreviations for the different combinations are summarized in Table 3.1. Hence, we investigated 90 different combinations of network architectures and optional parameters. It is clear that optimizing the hyperparameters (see Sec. 3.2.1) for each individual case is computationally infeasible, and therefore we optimized only the DN/x, CN/x, and UN/x networks. Then, those optimized hyperparameters are used for all other networks of the same architecture to systematically investigate the effect of the additional information provided to the NN as well as the representation of the input and output data. In most cases, the variation of the hyperparameters resulted in changes of the prediction accuracy on the order of the variation between the folds, thus typically less than the effect of the parameters listed in Table 3.1. Only in selected cases, we found that the NNs were too small to capture the relevant details or too large to cause overfitting.

Figure 3.3 shows the mean relative absolute error between the predicted and target pressures, $\langle L_{\text{MRAE}} \rangle = \langle |P - \hat{P}| / P \rangle$, for all investigated NNs at $k_B T = \varepsilon$.

| Abbreviation | $\beta\bar{U}(r)$ | $\exp[-\beta U(r)]$ | $F(r)$ | P_0 |
|--------------|-------------------|---------------------|--------|-------|
| x | ✗ | ✗ | ✗ | ✗ |
| c | ✓ | ✗ | ✗ | ✗ |
| e | ✗ | ✓ | ✗ | ✗ |
| f | ✗ | ✗ | ✓ | ✗ |
| i | ✗ | ✗ | ✗ | ✓ |
| cf | ✓ | ✗ | ✓ | ✗ |
| ci | ✓ | ✗ | ✗ | ✓ |
| ef | ✗ | ✓ | ✓ | ✗ |
| ei | ✗ | ✓ | ✗ | ✓ |
| fi | ✗ | ✗ | ✓ | ✓ |
| cfi | ✓ | ✗ | ✓ | ✓ |
| efi | ✗ | ✓ | ✓ | ✓ |

Table 3.1: List of abbreviations for the optional parameters of the networks.

The errorbars in Fig. 3.3 were calculated as the standard error of the mean over all four folds. For comparison, the mean relative absolute error between the target pressure P and the pressure P_0 estimated *via* B_2 was $\langle L_{\text{MRAE}} \rangle \approx 1.4$. In most cases, the NNs provide significantly more accurate predictions compared to P_0 , which is reasonable given that Eq. (3.3.1.3) is strictly valid only in the limit $\rho \rightarrow 0$. Further, the NNs that directly map $\beta U(r), \rho \mapsto \beta P$ perform worse than the NNs which first predict B_2^* or \mathcal{B}_n and then compute βP .

Using either the clipped potential $\beta\bar{U}(r)$ or $\exp[-\beta U(r)]$ as the input instead of $\beta U(r)$ typically led to better results and reduced overfitting in most of the cases. In contrast, including the forces did not improve the performance notably, but was even detrimental in some cases, as can be seen, for example, by comparing the performance of CN/fi and CN/i predicting B_2^* [see Fig. 3.3(e)]. We surmise that this behavior occurs because including the force does not provide any crucial additional information compared to the pair potential, while increasing the complexity of the network architecture and thus increasing the risk of overfitting. Providing βP_0 sometimes improved performance, but the effect was not as pronounced compared to the other parameters. For the UNs predicting B_2^* [Fig. 3.3(h)], including βP_0 resulted in a consistently lower accuracy on the validation data, which was likely due to overfitting as indicated by the small loss on the training set compared to the loss in the validation set throughout.

Based on the comparison shown in Fig. 3.3, we chose UN/c predicting \mathcal{B}_n

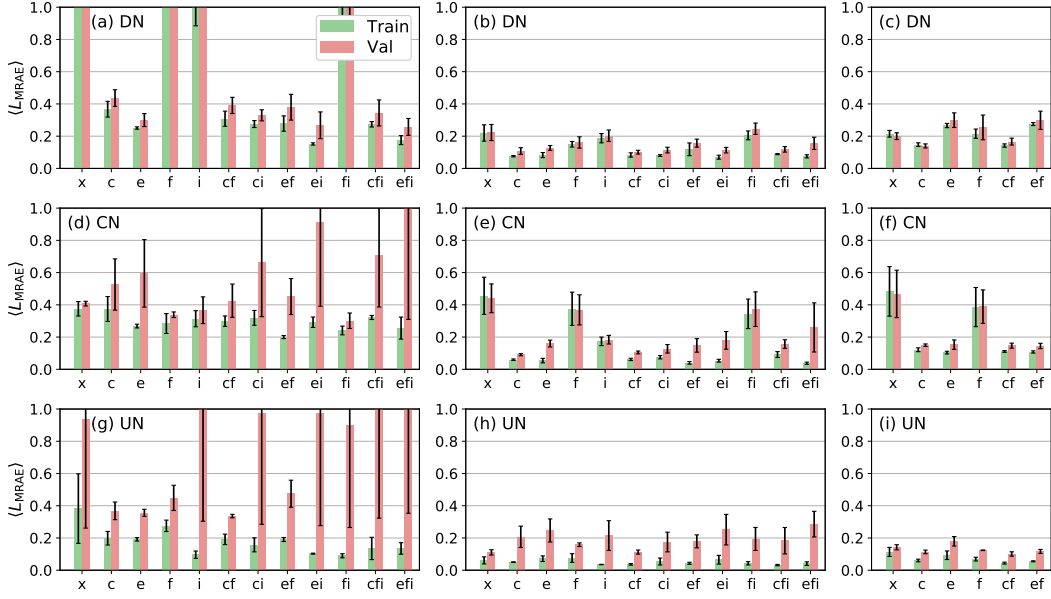


Figure 3.3: Mean relative absolute error $\langle L_{\text{MRAE}} \rangle$ between predicted pressures \hat{P} and target pressures P for different NN architectures and options at $k_{\text{B}}T = \varepsilon$. The panels are ordered from top to bottom as DN, CN and UN and from left to right as mapping to P , mapping to B_2^* and mapping to B_n .

as the final network because of the following reasons: The pressures predicted by this NN were among the most accurate with small mean relative absolute errors $\langle L_{\text{MRAE}} \rangle \approx 0.06 \pm 0.01$ and $\approx 0.11 \pm 0.01$ for the training and test set, respectively. Further, the mapping $\beta U(r) \mapsto \mathcal{B}_n$ should also generalize better to densities which were not part of the training set, as the predicted \mathcal{B}_n are explicitly density independent, with the only density dependence of $\beta \hat{P}$ coming from the virial expansion. Further, we used clipped potentials $\beta \bar{U}(r)$, which guaranteed that the values provided to the NN are bounded, leading to better generalizability and less overfitting. Forces were not included in the final network, because they did not always improve the prediction accuracy, but increased the model complexity and the risk of overfitting.

For the final benchmark, all data (except those in the test set) were used for training the NN. Figure 3.4 shows the pressure \hat{P} predicted by the UN/c network *vs.* the target pressure P at $k_{\text{B}}T = \varepsilon$. Here, we have also included the low-density estimate P_0 [see Eq. (3.3.1.3)]. The predicted pressures \hat{P} are very close to the target values P , with coefficients of determination $R^2 \approx 1.00$ for both the training and test set. In contrast, P_0 deviated strongly from P , especially for the systems at high density and/or with $P < 0$. This rather poor agreement with the target pressure is reflected in the coefficients of determination, which

was $R^2 \approx 0.172$.

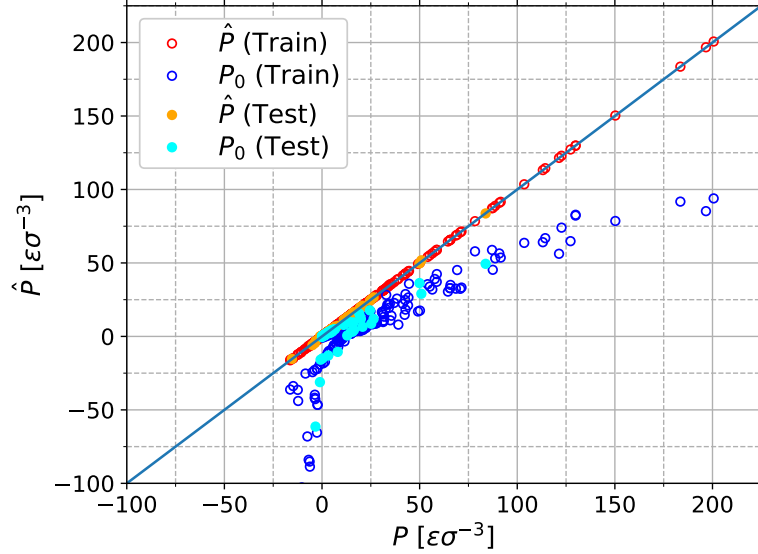


Figure 3.4: Predicted pressure \hat{P} from UN/c (red and orange symbols) and P_0 estimated from Eq. (3.3.1.3) (blue and cyan symbols) *vs.* the target pressure P at $k_B T = \varepsilon$. Open and filled symbols show results from the training and test set, respectively.

3.3.2 Potential prediction from pair distribution function

In this section we will construct NNs (see Fig. 3.5) for predicting (effective) pair potentials $U(r)$ from radial pair distribution functions $g(r)$ at a given, known particle number density ρ . We used either $g(r)$ as the input, or the (effective) pair potential in the low density limit

$$\beta U_0(r) := -\ln [g(r)]. \quad (3.3.2.1)$$

The reasoning behind the latter approach is to provide the NNs with a physically informed estimate which is valid for $\rho \rightarrow 0$, so that the NNs only need to learn perturbations to this solution at higher densities. Networks using $\beta U_0(r)$ as input are indicated with a “b” for Boltzmann inversion. Optionally, the NNs could optimize and output $\exp[-\beta U(r)]$ instead of $\beta U(r)$ (indicated by “e”), and/or include the force (indicated by “f”) in the loss function through Eq. (3.2.1.7).

The loss was always computed between the predicted, $\beta \hat{U}(r)$, and target pair potential, $\beta U(r)$, with $\beta = 1/\varepsilon$. To reduce the search space and avoid

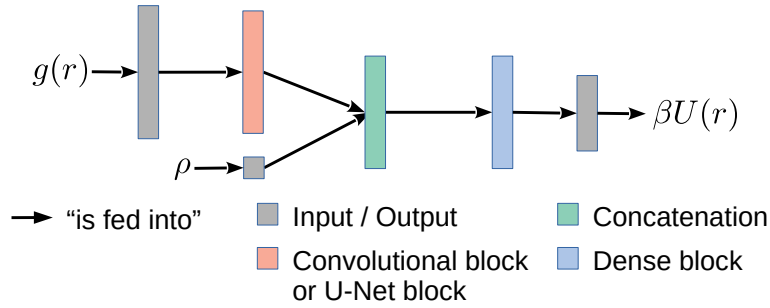


Figure 3.5: Schematic representation of the NNs used for predicting (effective) pair potentials $\beta U(r)$ from a radial pair distribution function $g(r)$.

overfitting, we disregarded parts of $\beta U(r)$ that exceeded $\beta U_{\text{cut}} = 20$, effectively yielding a sliced loss function. Hence, the predicted $\beta U(r)$ were not optimized in regions where $\beta U(r) > \beta U_{\text{cut}}$, allowing the NNs to set arbitrary values for the corresponding ranges.

Methods such as RMC or IBI instead optimize $\hat{g}(r)$ against the reference $g(r)$ by iteratively performing MD simulations with the predicted potential $\hat{U}(r)$ (see ESI for more details). Such an approach is, however, prohibitively time consuming as it would require additional MD simulations at each step of the optimization procedure. Further, minimizing the loss between $\hat{g}(r)$ and $g(r)$ is infeasible from a technical point of view, because the training of the NNs is based on gradient descent, which requires the (unknown) mapping $\beta U(r) \mapsto g(r)$. Thus, the difference between $g(r)$ and $\hat{g}(r)$ is not a viable input for the loss function for training the NNs, but it can be used as a final performance benchmark in selected instances (see Fig. 3.10 below).

| Abbreviation | $\beta U_0(r)$ input | force in loss | $\exp[-\beta \hat{U}(r)]$ output |
|--------------|----------------------|---------------|----------------------------------|
| x | ✗ | ✗ | ✗ |
| b | ✓ | ✗ | ✗ |
| e | ✗ | ✗ | ✓ |
| f | ✗ | ✓ | ✗ |
| be | ✓ | ✗ | ✓ |
| bf | ✓ | ✓ | ✗ |
| ef | ✗ | ✓ | ✓ |
| bef | ✓ | ✓ | ✓ |

Table 3.2: List of abbreviations for the optional parameters of the networks.

Again, we tested the DN, CN and UN architectures, and systematically an-

alyzed the influence of providing additional information and different representations of the input and output data. Figure 3.6 shows $\langle L_{\text{MSE}} \rangle$ and $\langle L_{\text{MAE}} \rangle$ between the predicted and the target potentials at $k_{\text{B}}T = \varepsilon$ for all investigated NNs and parameter combinations listed in Table 3.2. The errorbars were obtained through a 4-fold cross validation. The dotted lines correspond to the MSE (blue) and MAE (green) between $U(r)$ and $U_0(r)$ [note that we employed here the same clipping as for $\hat{U}(r)$].

As can be seen from Fig. 3.6, almost all investigated NNs provide more accurate predictions compared to $U_0(r)$. The predicted potentials $\hat{U}(r)$ from the NNs optimizing $\exp[-\beta\hat{U}(r)]$ have consistently larger $\langle L_{\text{MSE}} \rangle$ and $\langle L_{\text{MAE}} \rangle$ compared to the NNs which directly output $\beta\hat{U}(r)$. This discrepancy likely originates from the non-linearity of the transformation, which effectively modifies the importance of different regions of the potential. Figure 3.6 also shows that including the force in the loss function during the training stage improves the accuracy of the NNs (see Sec. 3.2.1). Further, the predicted potentials were significantly smoother when the force was included, as shown for one selected example in Fig. 3.7. Finally, we find that inputting $\beta U_0(r)$ instead of $g(r)$ did not improve the prediction significantly. Comparing the average losses $\langle L_{\text{MSE}} \rangle$ and $\langle L_{\text{MAE}} \rangle$ of the different network architectures for a given parameter set, we see that the CNs were most accurate in most cases for both the training and validation sets. In particular, the CN/f network had the smallest average losses and also small variations between the different folds. For final benchmarking, we therefore focus on the CN/f network, which was trained again with all data except those in the test set.

To evaluate the accuracy of this NN in more detail, we analyzed the probability density function of L_{MAE} for the training and test set, $p(L_{\text{MAE}})$, and compared it with $p(L_{\text{MAE}})$ obtained from using $U_0(r)$ (see Fig. 3.8). In both cases, $p(L_{\text{MAE}})$ can be fitted by an exponential decay $\propto \exp[-(L_{\text{MAE}}/\delta)]$, with δ quantifying the width of the distribution. For the CN/f network, $p(L_{\text{MAE}})$ had a rather narrow distribution with $\delta = 0.02 \varepsilon$ and $\delta = 0.03 \varepsilon$ for the training and test set, respectively, whereas $p(L_{\text{MAE}})$ was considerably wider for $U_0(r)$ with $\delta = 0.5 \varepsilon$ (the asymptotic standard error of the fits was about 10% in all cases). It should also be noted that the estimated $U_0(r)$ had several outliers with very large deviations up to $L_{\text{MAE}} \approx 8 \varepsilon$.

At this point, we also want to discuss the transferability of the predicted potentials $\beta\hat{U}(r)$. Given that $g(r)$ can vary significantly for the same $\beta U(r)$ at different particle number densities ρ , it is important to check whether $\beta\hat{U}(r)$ remains independent of ρ . Figure 3.9 shows a selected target potential $U(r)$ from the test set and the resulting $g(r)$ at four different ρ ($k_{\text{B}}T = \varepsilon$ fixed), highlighting the increasing deviations from ideal-like behavior [$g(r) = 1$] with increasing

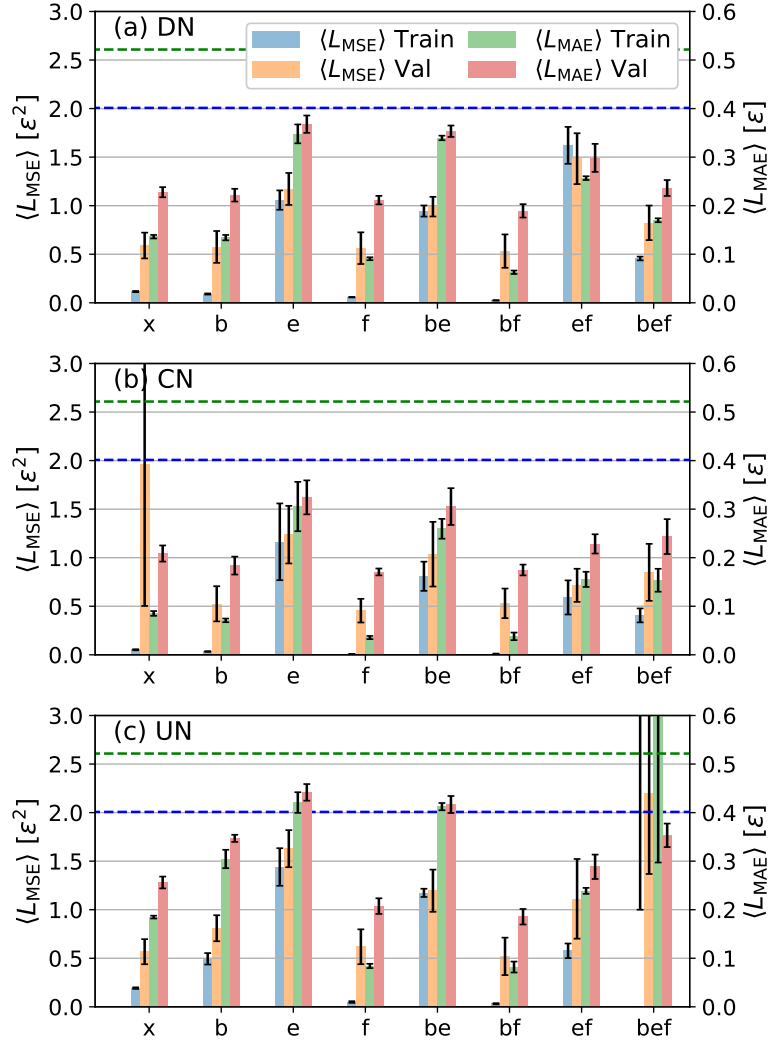


Figure 3.6: Comparison of $\langle L_{\text{MSE}} \rangle$ (left axis) and $\langle L_{\text{MAE}} \rangle$ (right axis) between the predicted, $\hat{U}(r)$, and target pair potential, $U(r)$, for (a) DNs, (b) CNs, and (c) UNs with different options (see Table 3.2). The dashed lines indicate $\langle L_{\text{MSE}} \rangle$ (blue) and $\langle L_{\text{MAE}} \rangle$ (green) between $U(r)$ and $U_0(r)$. Temperature fixed at $k_{\text{B}}T = \varepsilon$ in all cases.

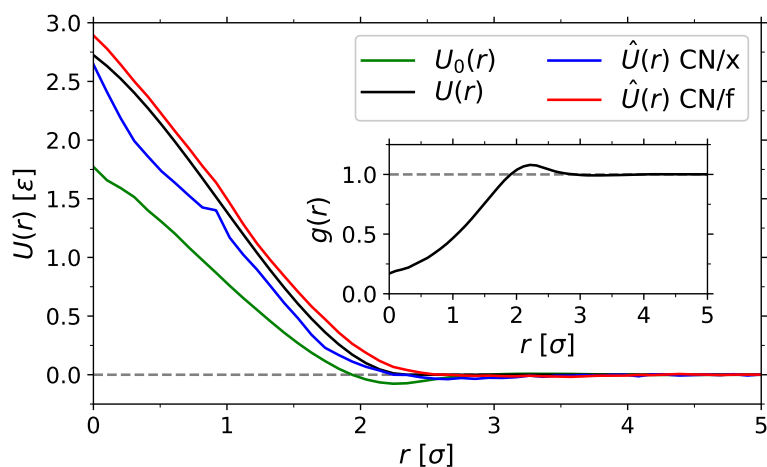


Figure 3.7: Selected target pair potential $U(r)$ from the validation set and predictions $\hat{U}(r)$ from CN/x and CN/f at $k_B T = \epsilon$. The estimated potential $U_0(r)$ is also shown for comparison. The inset shows the corresponding radial distribution function $g(r)$ at the employed density $\rho = 0.125 \sigma^{-3}$.

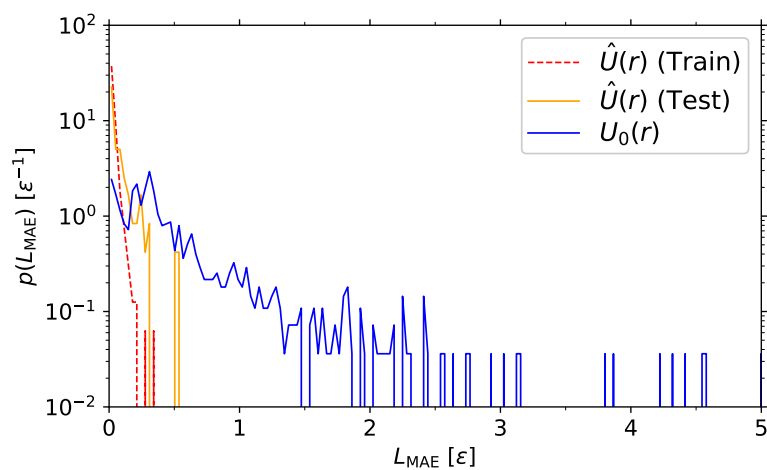


Figure 3.8: Probability density function of L_{MAE} between target $U(r)$ and predicted $\hat{U}(r)$ using CN/f, and between $U(r)$ and $U_0(r)$ at $k_B T = \epsilon$.

density. We have also included in Fig. 3.9(a) the predictions of $\hat{U}(r)$ from the CN/f network as well as $U_0(r)$ at different ρ . Indeed, the predictions $\hat{U}(r)$ are rather close to $U(r)$, and show only minor variations with ρ (as should be the case). In contrast, the estimate $U_0(r)$ provides a passable approximation of $\hat{U}(r)$ only at the lowest density $\rho = 0.125 \sigma^{-3}$, and the agreement with $U(r)$ gets significantly worse with increasing ρ .

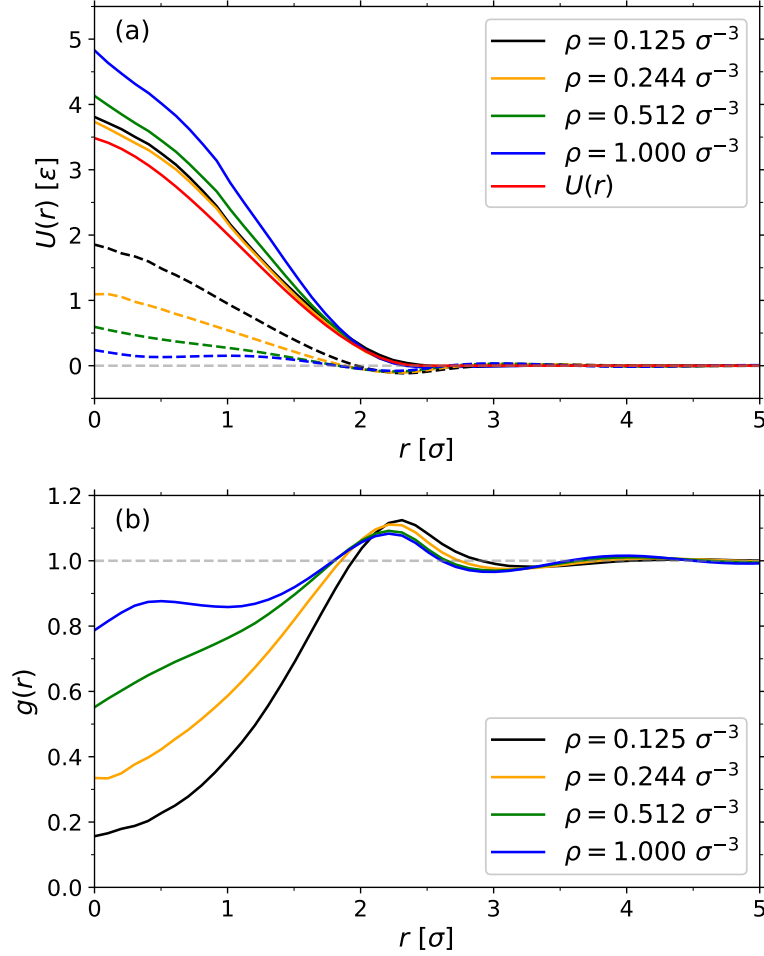


Figure 3.9: (a) Target potential $U(r)$ and predicted potentials $\hat{U}(r)$ from CN/f at four different particle number densities ρ , as indicated. The dashed lines show $U_0(r)$ at the same densities. (b) Radial pair distribution functions $g(r)$ from simulations with $U(r)$ at various ρ , as indicated. Temperature fixed at $k_B T = \epsilon$ in all cases.

To check whether the predicted $\beta \hat{U}(r)$ lead to similar pair distribution functions $\hat{g}(r)$ as the original $g(r)$, we ran MD simulations with $\hat{U}(r)$ (at temperature $k_B T = \epsilon$) for selected cases from the training and test set. To test the

performance of simple Boltzmann inversion, we also ran MD simulations with $U_0(r)$ and calculated the corresponding $g_0(r)$. Figure 3.10 shows the comparison between $g(r)$, $\hat{g}(r)$ and $g_0(r)$, where we have also included the employed pair potentials $U(r)$, $\hat{U}(r)$, and $U_0(r)$ as insets of the corresponding plots. In all considered cases, $\hat{g}(r)$ is much closer to $g(r)$ than $g_0(r)$, especially for distances $r \gtrsim \sigma$. The systems shown in Fig. (3.10)(a,b) have been run with pair potentials from the training set at densities $\rho = 0.244 \sigma^{-3}$ and $0.512 \sigma^{-3}$, respectively. These cases are rather interesting, because $U(r)$ and $\hat{U}(r)$ look almost identical, but the resulting $g(r)$ and $\hat{g}(r)$ show some deviations for $r \lesssim \sigma$. The estimated potentials $U_0(r)$ differ drastically from $U(r)$, but the resulting $g_0(r)$ for $\rho = 0.244 \sigma^{-3}$ is relatively close to the target one. At higher density $\rho = 0.512 \sigma^{-3}$, however, the simulations performed with $U_0(r)$ lead to a distinct clustering of particles, while the simulations with $U(r)$ and $\hat{U}(r)$ lead to an almost flat, gas-like $g(r)$ with weak oscillations. Thus, in this case, $U_0(r)$ is not even a good starting point for iterative optimization methods like IBI, as demonstrated in Figs. S1 and S3 of the ESI, where we performed 200 optimization iterations¹¹⁵ starting from $U_0(r)$. Even when initializing the IBI procedure with $\hat{U}(r)$, which is rather close to $U(r)$, IBI pushes the prediction away from $U(r)$.

Figure (3.10)(c,d) shows results for cases from the test set (same systems as in Fig. 3.7) at two different densities. At low density $\rho = 0.244 \sigma^{-3}$, the agreement between $g(r)$ and $\hat{g}(r)$ is almost perfect, while $g_0(r)$ replicates the qualitative correct trends. For the denser systems $\rho = 0.512 \sigma^{-3}$, both $\hat{g}(r)$ and $g_0(r)$ exhibit a qualitatively similar deviation from the target radial distribution function, but the predicted $\hat{U}(r)$ is much closer to the target pair potential $U(r)$ compared to $U_0(r)$. The IBI procedure converges if either $U_0(r)$ or $\hat{U}(r)$ is used as the starting point, but convergence is achieved much faster for $\hat{U}(r)$ (after 2 iterations) than for $U_0(r)$ (after roughly 20 iterations), as shown in Fig. S3.

As a final test of our methodology, we used the predicted pair potentials $\hat{U}(r)$ from the CN/f networks as an input of the UN/c networks developed in Sec. 3.3.1 to calculate the pressure \hat{P} from the radial pair distribution function $g(r)$. Figure 3.11 shows \hat{P} vs. P at $k_B T = \varepsilon$, demonstrating rather good agreement for the training set ($R^2 \approx 0.971$). The predictions for the test set have a significantly lower accuracy ($R^2 \approx 0.461$), but the overall trends are captured in most cases (excluding the outlier at $P \approx 80 \varepsilon / \sigma^3$ from the test set leads to $R^2 \approx 0.802$). For comparison, we have also included the pressure P_0 , which we estimated by computing $U_0(r)$ from $g(r)$. These pressures are almost always much smaller than the target pressure, leading to $R^2 \approx 0.0192$.

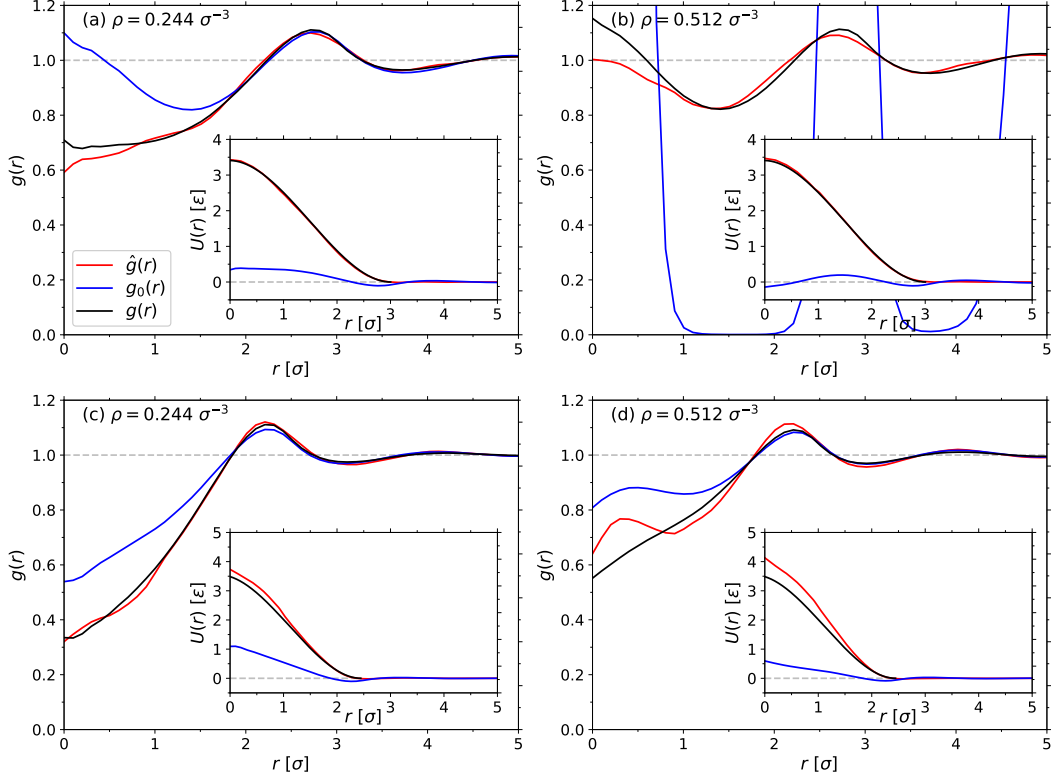


Figure 3.10: Comparison between radial pair distribution functions $g(r)$, $\hat{g}(r)$, and $g_0(r)$ computed from MD simulations performed with $U(r)$, $\hat{U}(r)$ and $U_0(r)$, respectively. The corresponding potentials have been plotted in the insets. Temperature fixed at $k_B T = \varepsilon$ in all simulations. Cases (a,b) are from the training set with (a) $\rho = 0.244 \sigma^{-3}$ and (b) $\rho = 0.512 \sigma^{-3}$, while cases (c,d) are from the test set with (c) $\rho = 0.244 \sigma^{-3}$ and (d) $\rho = 0.512 \sigma^{-3}$. The L_{MSE} between the original $g(r)$ and $\hat{g}(r)$ are (a) 1.4×10^{-4} (5.1×10^{-3}), (b) 5.0×10^{-4} (5.7×10^{-3}), (c) 2.9×10^{-5} (2.3×10^{-3}), and (d) 5.0×10^{-4} (2.4×10^{-3}). The numbers in parentheses indicate L_{MSE} between $g(r)$ and $g_0(r)$.

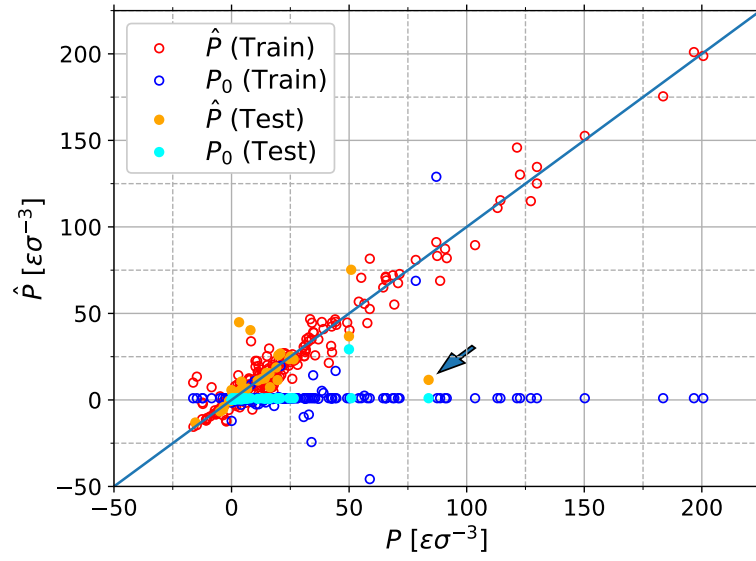


Figure 3.11: Predicted pressure \hat{P} from UN/c (red and orange symbols) and P_0 estimated from Eq. (3.3.1.3) (blue and cyan symbols) *vs.* the target pressure, P , using the potentials $\hat{U}(r)$ and $U_0(r)$, respectively. Open and filled symbols show results from the training and testing set, respectively. The outlier in the test set for the calculation of \hat{P} has been indicated with an arrow. Temperature fixed at $k_B T = \epsilon$ in all cases.

3.3.3 Use case examples

Prediction of the EOS for Lennard-Jones and Gaussian particles

To further test the applicability of our approach, we employed the UN/c to predict the EOS of Lennard-Jones particles at temperatures below and above the critical temperature $k_B T \approx 1.31 \varepsilon$.¹¹⁶ The pressures have been predicted from a truncated ($r_{\text{cut}} = 4.0 \sigma$) and shifted Lennard-Jones potential U_{LJ} . The resulting \hat{P} are plotted in Fig. 3.12(a) together with the pressure P from the modified Benedict-Webb-Rubin (MBWR) EOS determined by Johnson *et al.*¹¹⁷ At $k_B T = 1.4 \varepsilon$, \hat{P} match the reference data almost perfectly, except for some deviations at high densities $\rho \approx 1.0 \sigma^{-3}$ near the liquid-solid coexistence line. The deviations between \hat{P} and P become more pronounced with decreasing temperature, with \hat{P} being systematically above P , but remaining well below the Carnahan-Starling EOS for the fluid phase of hard spheres.¹¹⁸ We quantified the accuracy of the NN by computing the MAE between \hat{P} and P , finding that the error increased from $L_{\text{MAE}} = 0.098 \varepsilon / \sigma^3$ at $k_B T = 1.4 \varepsilon$ to $L_{\text{MAE}} = 0.75 \varepsilon / \sigma^3$ at $k_B T = 0.8 \varepsilon$. Despite the differences at lower temperatures, the accuracy of the predicted pressures \hat{P} is still satisfactory, considering that we did not explicitly include U_{LJ} in our training set. Instead, our procedure for generating pair potentials $U(r)$ (see Sec. 3.2.2) produced only few $U(r)$ that were similar to U_{LJ} . The inset of Fig. 3.12(a) shows three selected potentials from our training set that are close to U_{LJ} with $\varepsilon_{\text{LJ}} = 0.8 \varepsilon$, $\varepsilon_{\text{LJ}} = 0.92 \varepsilon$, and $\varepsilon_{\text{LJ}} = \varepsilon$ ($\sigma_{\text{LJ}} = \sigma$ in all cases). Further, training data were generated only at few densities, which cover a small portion of the density range that was sampled for the EOS plotted in Fig. 3.12(a).

To improve the accuracy of our NN for this use case, we performed 28 MD simulations of particles interacting *via* U_{LJ} , covering the range $0.6 \varepsilon \leq k_B T \leq 1.4 \varepsilon$ and $0.05 \sigma^{-3} \leq \rho \leq 1.0 \sigma^{-3}$, and included the additional data in our training set. We stress that, in accordance with good machine learning practices, we did not evaluate our model on the test set again to avoid leakage. The newly predicted pressures \hat{P} from the expanded UN/c are much closer to the reference pressures P from the MBWR EOS than our initial predictions, with MAE values ranging between $L_{\text{MAE}} = 0.16 \varepsilon / \sigma^3$ and $0.24 \varepsilon / \sigma^3$. Given the small number of simulations with U_{LJ} in this new training set, this improvement is rather remarkable, indicating that even a small number of examples from a new class of potentials can lead to a better generalization of our machine learning approach. As a comparison, Johnson *et al.* used data from 182 different simulations (80 in the considered temperature and density range) to parameterize their EOS.¹¹⁷ If we include all their data in the training set, then the predicted pressures further approach the target values (MAE ranging between $L_{\text{MAE}} = 0.082 \varepsilon / \sigma^3$

and $0.19 \varepsilon/\sigma^3$), as shown in Fig. 3.12(b).

As a second test, we predicted the EOS for a system of particles interacting *via* the purely repulsive Gaussian pair potential $U(r) = \varepsilon \exp[-(r/\sigma)^2]$. Figure 3.13 shows both the predicted pressures \hat{P} and reference pressures P from MD simulations as functions of ρ for three selected temperatures. In all cases, \hat{P} and P are in excellent agreement over the entire investigated density regime. In contrast, the pressure estimates P_0 based on the second virial coefficient [see Eq. (3.3.1.3)] are systematically smaller than P (not shown here), and the deviations between P and P_0 grow with increasing density.

Determining effective potentials in bulk polymer melts

As a second use case example, we employed the CN/f networks developed in Sec. 3.3.2 to determine the effective pair interactions between coarse-grained polymers in a melt. The microscopically resolved (MR) reference system consisted of $N_p = 2000$ homopolymer chains with $N = 20$ monomers each. The monomer-monomer interactions were given by a truncated ($r_{\text{cut}} = 5.0 \sigma$) and shifted Lennard-Jones potential, while bonded interactions were included *via* the finitely extensible nonlinear elastic (FENE) potential⁹ with spring constant $k = 30 \varepsilon/\sigma^2$ and maximum bond extension $r_0 = 1.5 \sigma$.⁸ The MR simulations have been conducted at $k_B T = \varepsilon$ in a cubic simulation box with an edge length of $L = 35.15 \sigma$ and periodic boundary conditions applied to all Cartesian directions. In the coarse-grained simulations, each polymer should be represented by a single spherical particle located at the polymer's center of mass.

To determine the effective potential between these coarse-grained particles, we computed the radial pair distribution function $g(r)$ from the MR simulations and fed it into our NN with polymer density $N_p/L^3 \approx 0.046 \sigma^{-3}$. Figure 3.14 shows $g(r)$ from the MR simulations as well as $\hat{g}(r)$ from the coarse-grained simulations with the effective pair potential $\hat{U}(r)$ from the CN/f network. These structural data are in rather good agreement, except for a slightly more pronounced overlap of polymers at short distances in the coarse-grained simulations. We quantified the difference between $g(r)$ and $\hat{g}(r)$ through the mean squared error, $L_{\text{MSE}} = 0.0138$. We then optimized the effective pair potential using IBI, with 200 iterations of 10^6 timesteps each ($\Delta t = 0.001 \tau$). The difference between $g(r)$ and $\hat{g}^i(r)$, with superscript i denoting the iteration of the IBI procedure, decreased to $L_{\text{MSE}} = 0.00261$ and 0.00249 after 10 and 200 iterations, respectively. As can be seen in the inset of Fig. 3.14, the effective pair potential became more repulsive during the iterative optimization, which reduced the overlap of coarse-grained particles. This example demonstrates how our machine learning approach can be used to quickly generate accurate coarse-grained representations, which then can be further optimized using

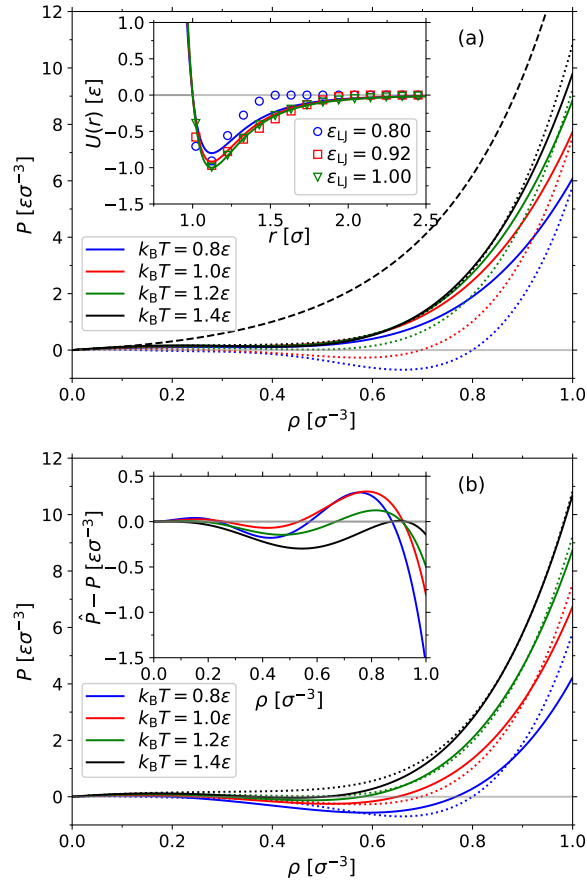


Figure 3.12: (a) Pressure P vs. particle number density ρ for LJ particles at four different temperatures, as indicated. Solid lines show the predictions from UN/c, while dotted lines show the reference from the MBWR EOS.¹¹⁷ The dashed black line corresponds to the Carnahan-Starling EOS. Inset: Three selected pair potentials from the training set (symbols) and the corresponding LJ potentials with strength ϵ_{LJ} (lines). (b) Same as (a) but for an updated UN/c that contains additional data obtained from Ref. [117] in the training set. Inset: Pressure difference $\hat{P} - P$ vs. ρ for the curves shown in the main panel.

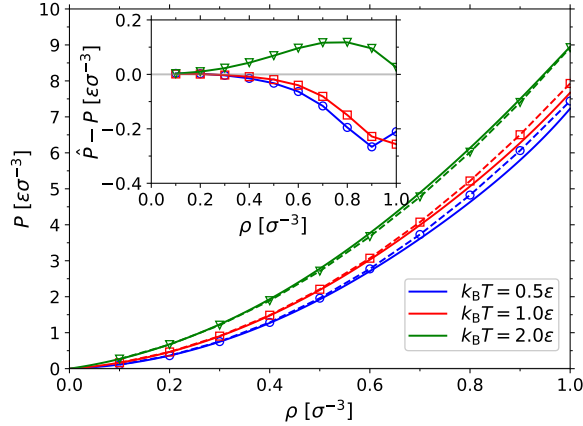


Figure 3.13: P vs. ρ for particles interacting through $U(r) = \varepsilon \exp[-(r/\sigma)^2]$ at three different temperatures, as indicated. Solid lines show the predicted pressures \hat{P} from UN/c , while symbols connected by dotted lines show the reference pressure P from MD simulations. Inset: Pressure difference $\hat{P} - P$ vs. ρ for the curves shown in the main panel.

traditional (iterative) techniques.

3.4 Conclusions

Artificial neural networks (NNs) were developed for predicting the equation of state for a given pair potential $U(r)$ at density ρ and temperature T , and for predicting (effective) pair potentials from structural information based on the radial distribution function $g(r)$. We investigated how the representation of the input and output data as well as additional information (*e.g.*, the forces) influence the prediction accuracy. For both tasks, a key preprocessing step was to limit the input and/or output range by capping the pair potential $\beta U(r)$ so that it could not reach arbitrarily large values. For predicting the pressure P , NNs were tested that directly map the input pair potential $\beta U(r)$ to βP , and NNs that first predict (effective) virial coefficients from $\beta U(r)$ which are then used to compute βP . The latter strategies resulted in rather accurate predictions, which were much closer to the target pressure ($R^2 \approx 1.00$) compared to the virial expansion derived in the low-density limit. For predicting (effective) pair potentials, the accuracy of the predictions improved significantly when also the derivatives of the potentials were provided in the loss, since more correlations were taken into account. As a result, the predicted potentials became smoother and were significantly closer to the target potentials compared to

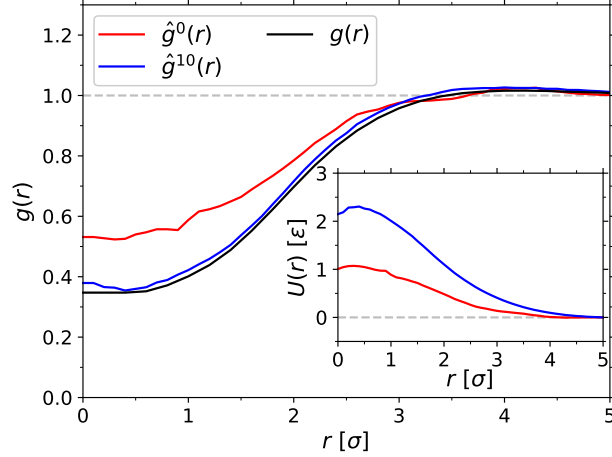


Figure 3.14: Radial pair distribution function between the centers of mass of polymers from microscopically resolved [$g(r)$] and coarse-grained [$\hat{g}(r)$] simulations. The data for $\hat{g}^0(r)$ are from MD simulations using $\hat{U}^0(r)$ predicted by the CN/f, while $\hat{g}^{10}(r)$ are from MD simulations with $\hat{U}^{10}(r)$ after 10 iterations of Boltzmann inversion. The inset shows $\hat{U}^0(r)$ and $\hat{U}^{10}(r)$.

results obtained through simple Boltzmann inversion. To test the viability of our approach for practical applications, we employed the NNs to predict the equation of state of Lennard-Jones particles, achieving good agreement with literature data after expanding the training set. Further, we successfully applied our NNs to determine the effective pair interactions for coarse-graining polymers in a melt.

Our NN approach produced reasonably accurate predictions, which could be used as a starting point for further optimizations *via* other (iterative) methods until the desired accuracy is achieved. Such a combined approach could drastically cut down the computational cost and development time for coarse-graining applications, and it could also be a useful tool for inverse problems in materials discovery. There is, of course, still room for future improvements: The training data used in this work covered only a subset of all conceivable systems, and adding more data could significantly improve the prediction accuracy, especially if the additional data covers new classes of potentials. Further, using a diverse set of training data will naturally reduce the risk of overfitting, and also expand the applicability of our approach to a wider range of systems. The employed machine learning strategy can also be extended beyond pressure prediction in the canonical ensemble and structure-based coarse-graining. For example, the same strategies and existing data could be used to train NNs for predicting the density and total energy from the pair potential in the isothermal-isobaric ensemble, or new data from simulations in the grand canonical ensemble.

ble could be included to predict the chemical potential of a system. One could also include the pressure in the coarse-graining procedure to generate models that reproduce the pressure and compressibility of the target systems more accurately.^{103,119–121} Improving these aspects of our machine learning approach is an ongoing effort, and we welcome the help of the scientific community to reach these goals.

Supporting Material

Plots of all potentials and radial distribution functions used for training and testing; Additional information on theoretical background; Schematics of final network architectures

Acknowledgements

We thank M. R. Khadilkar for fruitful discussions. This work was funded by the German Research Foundation (DFG) through project number 233630050 - TRR 146. AN further acknowledges financial support provided by the DFG through project number NI 1487/2-1 and NI 1487/2-2. Computing time was granted on the supercomputer Mogon at Johannes Gutenberg University Mainz (www.hpc.uni-mainz.de).

Data Availability Statement

The data that supports the findings of this study are available within the article and its supplementary material.

CHAPTER 4 --- Ultra-Coarse-Graining of Homopolymers in Inhomogeneous Systems

The following chapter has been published in word as:¹²²

Fabian Berressem, Christoph Scherer, Denis Andrienko and Arash Nikoubashman, "Ultra-Coarse-Graining of Homopolymers in Inhomogeneous Systems", *Journal of Physics: Condensed Matter* **33**, 254002 (2021) <https://doi.org/10.1088/1361-648X/abf6e2>

Using the CRediT author statement,⁶⁹ the contributions to this work are as follows:

- Fabian Berressem:** Conceptualization (equal); Methodology (lead); Software (lead); Validation (equal); Formal analysis (lead); Investigation (lead); Data curation (lead); Writing - original draft (lead); Writing - review & editing (equal); Visualization (lead)
- Christoph Scherer:** Conceptualization (equal); Methodology (equal); Software (supporting); Validation (equal); Formal analysis (equal); Investigation (equal); Data curation (equal); Writing - original draft (equal); Visualization (equal); Writing - review & editing (equal);
- Denis Andrienko:** Conceptualization (equal); Validation (supporting); Formal analysis (supporting); Resources (supporting); Writing - original draft (equal); Writing - review & editing (supporting); Supervision (supporting); Project administration (lead); Funding acquisition (lead)
- Arash Nikoubashman:** Conceptualization (equal); Methodology (supporting); Software (supporting); Validation (supporting);

Formal analysis (supporting); Investigation (supporting);
Resources (lead); Writing - original draft (equal);
Writing - review & editing (equal); Supervision (lead);
Project administration (lead); Funding acquisition (lead)

Abstract

We develop coarse-grained (CG) models for simulating homopolymers in inhomogeneous systems, focusing on polymer films and droplets. If the CG polymers interact solely through two-body potentials, then the films and droplets either dissolve or collapse into small aggregates, depending on whether the effective polymer-polymer interactions have been determined from reference simulations in the bulk or at infinite dilution. To address this shortcoming, we include higher order interactions either through an additional three-body potential or a local density-dependent potential. We parameterize the two- and three-body potentials *via* force matching, and the local density-dependent potential through relative entropy minimization. While the CG models with three-body interactions fail at reproducing stable polymer films and droplets, CG simulations with a local density-dependent potential are able to do so. Minor quantitative differences between the reference and the CG simulations, namely a slight broadening of interfaces accompanied by a smaller surface tension in the CG simulations, can be attributed to the deformation of polymers near the interfaces, which cannot be resolved in the CG representation, where the polymers are mapped to spherical beads.

4.1 Introduction

Coarse-graining is a systematic approach to reduce the number of degrees of freedom for building a simplified model of a system which reproduces its essential physical properties. The major advantage of coarse-grained (CG) models is that they provide access to longer simulation time- and length-scales, by reducing the number of interaction sites and introducing softer interaction potentials which accelerate the dynamics. Numerous CG models have been developed for simulating, *e.g.*, polymer melts,^{72,123,124} organic solvents,^{125–127} lipid membranes,^{128,129} conjugated polymers,^{130–135} peptides,¹³⁶ surfactants,¹³⁷ and proteins.¹³⁸

CG potentials can be viewed as a projection of a many-body potential of mean force onto a CG force field.^{42,139} This projection is, however, not unique, as it depends on the thermodynamic or structural properties which should be preserved. Coarse-graining based on, *e.g.*, reproducing entropy,^{36,52,140,141} forces,^{42,50,139,142} or structure^{46,143} usually leads to distinct CG models of the system as the projection to a CG potential is in general not unique. In the limit of a complete set of CG basis functions, the different techniques will indeed lead to the same true many-body CG potential of mean force. However, this equivalency does not apply to practical cases of relevance with a limited CG basis set, as structure- and relative entropy-based CG methods minimize a different functional than force matching methods.^{42,52} Further, although there is a one-to-one relation between the pair structure and two-body potentials,⁴⁵ such associations generally do not hold for higher order structural correlations and potentials. As a result, different parametrization schemes can lead to different CG potentials,¹⁴⁴ and therefore CG models cannot represent all features of the original reference systems (representability problem). Further, CG models are typically optimized at one specific state point, *e.g.*, a certain temperature and/or pressure, and are therefore not necessarily suited for studying the same system at a different state point (transferability problem).^{101,145} Finally, the CG force fields should be computationally efficient to compensate for the loss in molecular details.

While a fairly coherent understanding of bottom-up coarse-graining based on structural correlations or inter-atomic forces has been established for *homogeneous* liquids in the *bulk*,⁴² one is often interested in phenomena taking place at interfaces, which are much less understood. Typical examples are wetting films or droplets and aqueous/organic interfaces in biological cells.^{146,147} These systems are intrinsically inhomogeneous, either in density or other local descriptors. Apart from these examples, CG models are particularly sensitive to local density fluctuations, even in bulk systems, due to their softer interaction potentials and smaller number of particles.^{148–150} Local inhomogeneity imposes extra demands on the CG model, in particular its tolerance to density variations.^{151–153} Higher-order many-body expansions^{107,154–158} and an explicit density dependence of interaction potentials^{101,153,159–162} are two straightforward approaches that can improve transferability of CG models. These improvements have, however, different accuracy and computational overhead.

The aim of this work is to compare the accuracy and efficiency of two CG models for simulating homopolymers in inhomogeneous systems, such as thin films or droplets. These systems are characterized by large surface-to-volume ratios and strong density variations near the polymer-solvent interface. The first model incorporates three-body Stillinger-Weber basis functions, while the

second has an explicit local density dependence. We show that the three-body expansion is computationally demanding and does not lead to stable interfaces. In contrast, the local density potential is capable of reproducing properties of inhomogeneous systems without a significant computational overhead.

The remainder of this manuscript is organized as follows: In Sec. 4.2.1 we present our reference systems, while we have summarized the employed coarse-graining methods in Sec. 4.2.2. The resulting two- and many-body interactions of the CG models are discussed in Secs. 4.3.1-4.3.3, and the properties of the CG simulations is analyzed and compared to the reference simulations in Sec. 4.3.4. Section 4.4 provides a brief summary of our main findings and an outlook on planned extensions of our model.

4.2 Methods

4.2.1 Microscopically resolved reference systems

The microscopically resolved (MR) reference systems consisted of N_p homopolymers dispersed in an implicit solvent. Polymers were represented by a generic bead-spring model with $N = 20$ monomers per chain. Non-bonded interactions between the monomers were modeled using the Lennard-Jones (LJ) potential:

$$U^{\text{LJ}}(r_{ij}) = \begin{cases} 4\varepsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right], & r_{ij} < r_c \\ 0, & r_{ij} \geq r_c \end{cases} \quad (4.2.1.1)$$

with radial distance r_{ij} between particles i and j , interaction strength $\varepsilon = k_B T$, and bead diameter σ . The potential was truncated at the cutoff radius of $r_c = 5\sigma$.

Polymer bonds were modeled through the finitely extensible nonlinear elastic (FENE) potential:⁹

$$U^{\text{FENE}}(r_{ij}) = \begin{cases} -\frac{1}{2} k r_0^2 \ln [1 - (r_{ij}/r_0)^2], & r_{ij} < r_0 \\ \infty, & r_{ij} \geq r_0 \end{cases} \quad (4.2.1.2)$$

with spring constant $k = 30 k_B T / \sigma^2$ and maximum bond extension $r_0 = 1.5\sigma$.⁸

All MD simulations were conducted in the \mathcal{NVT} ensemble (unless stated otherwise explicitly), $\mathcal{N} = N_p N$ being the total number of monomers in the system. The temperature was kept constant at $T = 1.0 \varepsilon / k_B$ through a Langevin thermostat with friction coefficient $\xi = 1.0 m / \tau$, where m is the monomer

mass, and $\tau = \sqrt{m\sigma^2/(k_B T)}$ is the unit of time. The equations of motion were integrated using a Verlet scheme with time step $\Delta t = 0.001 \tau$. All simulations were conducted in cubic simulation boxes with edge length L and periodic boundary conditions in all three Cartesian directions.

To characterize the polymer properties in homogeneous systems, we simulated a bulk polymer melt with $N_p = 2000$ chains. The melt was first equilibrated in the \mathcal{NPT} ensemble at a pressure of $P = 0 k_B T/\sigma^3$, resulting in an average monomer number density of $\rho_b = 0.921 \sigma^{-3}$ (all quantities extracted from bulk simulations will be denoted by a subscript “b”, while the subscript “0” marks quantities obtained or parameterized at infinite dilution). Once the system reached equilibrium, we switched to the \mathcal{NVT} ensemble using a cubic simulation box with $L = 35.15 \sigma$, and simulated for 2×10^7 time steps, saving configurations and taking measurements every 1000 time steps. We characterized the conformation of the polymers through the average radius of gyration tensor:

$$\mathbf{G} = \frac{1}{N} \sum_{i=1}^N \langle \Delta \mathbf{r}_i \otimes \Delta \mathbf{r}_i \rangle \quad (4.2.1.3)$$

where $\Delta \mathbf{r}_i$ is the vector from the polymer center of mass to monomer i , and \otimes indicates the dyadic product. The radius of gyration is then $R_g = \sqrt{G_{xx} + G_{yy} + G_{zz}}$, with $R_{g,b} \approx 2.2 \sigma$ in the bulk simulations.

Thin films were constructed by initially placing the polymers in a slab at the center of the simulation box, so that the normal vectors of the polymer-solvent interfaces lied parallel to the z -axis. The polymer droplets were created similarly by initializing the polymers close to each other in the box center. The size of the simulation boxes was chosen sufficiently large to prevent nonphysical self-interactions and the coalescence of the films/droplets with their periodic images (see Table 4.2). Simulations were then performed in the \mathcal{NVT} ensemble at $T = 1.0 \varepsilon/k_B$. This state point falls inside the two-phase coexistence region of the system,¹⁶³ so that the polymers separated into a high and low density phase that coexisted in the same simulation box. Simulation snapshots of typical films and droplets from the MR simulations are shown in Fig. 4.1. We determined the film thicknesses H and droplet diameters $2R$ from the full width at half maximum of the monomer density profiles [see Fig. 4.2(a)]. The resulting H and R values are summarized in Table 4.2, and they are in good agreement with the estimates $H = \mathcal{N}/(L^2 \rho_b)$ and $R = [3\mathcal{N}/(4\pi \rho_b)]^{1/3}$, respectively, which assume a homogeneous monomer density inside the films/droplets and a perfectly sharp polymer-solvent interface.

Figure 4.2(a) shows the monomers number density along the film normal, $\rho(z)$, for the system containing $N_p = 1245$ homopolymers (the data for the thinner films look qualitatively similar and have been omitted for brevity). Since

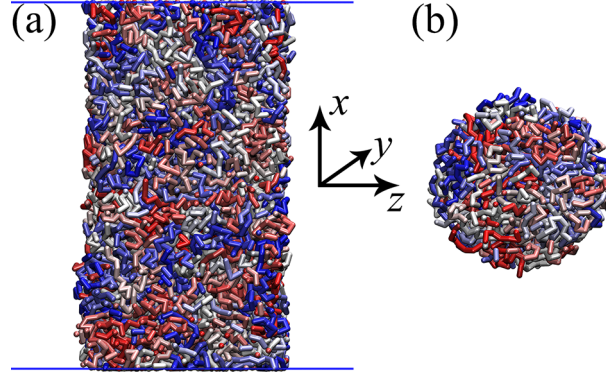


Figure 4.1: Simulation snapshots of (a) a thin film with thickness $H = 8 R_{g,b}$, and (b) a droplet with radius $R = 4 R_{g,b}$ from the MR reference simulations. The chains are colored differently for better distinction. Snapshots have been rendered using Visual Molecular Dynamics v.1.9.3.¹⁶⁴

| System | N_p | L [σ] | H or R [σ] |
|---------|-------|------------------|-------------------------|
| Film | 365 | 28.8 | 9.6 |
| Film | 728 | 33.2 | 14.4 |
| Film | 1245 | 37.6 | 19.2 |
| Droplet | 143 | 31.6 | 8.5 |
| Droplet | 482 | 40.4 | 13.0 |
| Droplet | 1142 | 49.2 | 17.6 |

Table 4.2: Number of homopolymers N_p in polymer films and droplets, with measured thickness H and radius R , respectively.

the simulated temperature was far below the critical temperature of this system ($T_c \approx 2.65 \varepsilon/k_B$ ¹⁶³), all polymers were part of the thin film (the density in the “vapor” phase was strictly zero in our simulations). The density profile has a flat plateau with density $\rho = 0.922 \sigma^{-3}$, which is in excellent agreement with the value measured in the bulk systems ($\rho_b = 0.921 \sigma^{-3}$). Measurement uncertainties of the density profiles were estimated from the standard error of the mean between ten subdivided blocks of the data. Note that we did not determine the (apparent) width of the polymer-solvent interfaces, as this measurement would only be meaningful when the dependence on the lateral box dimensions is analyzed to account for the broadening due to capillary waves.¹⁶⁵

The density profile of the polymer centers of mass, $\rho_p(z)$, is also flat in the bulk-like interior of the film, but has two distinct peaks at $z = \pm 8 \sigma$, indicating a distinct surplus of polymers near the polymer-solvent interfaces [Fig. 4.2(a)].

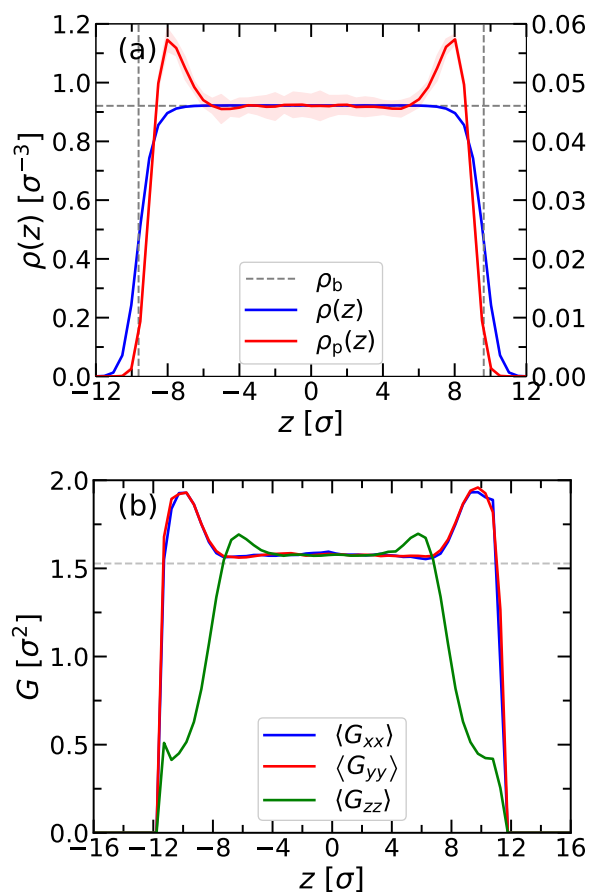


Figure 4.2: (a) Number density profiles of monomers (ρ , left axis) and polymer centers of mass (ρ_p , right axis) as functions of z . The horizontal dashed line indicates the monomer number density in the bulk system. The vertical dashed lines indicate the positions where $\rho(z)$ drops to half of its maximum value. (b) Diagonal components of the radius of gyration tensor, $G_{\alpha\alpha}$, as functions of z . The horizontal dashed line indicates the value of $G_{\alpha\alpha}$ in the bulk simulation. Data shown for a film with thickness $H \approx 8 R_{g,b}$, consisting of $N_p = 1245$ homopolymers.

This local excess of polymers can be rationalized by considering the conformation of the individual chains, which we have characterized *via* the diagonal components of the radius of gyration tensor [Fig. 4.2(b)]. The polymers are essentially isotropic in the central region of the film, but they assume an oblate ellipsoidal shape near the edges of the film, with $G_{zz} < G_{yy} = G_{xx}$ and a maximum aspect ratio of about $G_{xx}/G_{zz} \approx 4$.

Finally, we determined the surface tension of the thin films from the anisotropy of the pressure tensor:

$$\gamma = \frac{L_z}{2} \left\langle P_{zz} - \frac{P_{xx} + P_{yy}}{2} \right\rangle \quad (4.2.1.4)$$

where $P_{\alpha\alpha}$ denotes the diagonal components of the instantaneous pressure tensor, which was computed from the Clausius virial equation. The factor of $1/2$ in Eq. (4.2.1.4) is due to the presence of two interfaces [see Fig. 4.1(a)]. Here, we find a value of $\gamma = 1.42 k_B T / \sigma^2$ for all three investigated film thicknesses.

4.2.2 Coarse-graining procedure

In our CG model, an entire homopolymer chain is represented by a single spherical particle, as depicted in Fig. 4.3. We tested three different approaches, using (i) only two-body interactions, (ii) a combination of two-body and three-body interactions, and (iii) a combination of two-body and (mean-field) many-body interactions. In the following, we briefly discuss these interactions and how they have been parameterized. The resulting CG potentials are then presented in Sec. 4.3.

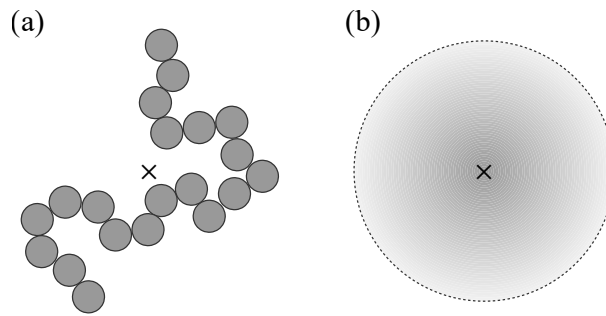


Figure 4.3: Schematic representation of (a) a microscopically resolved polymer chain with $N = 20$ monomers, and (b) the coarse-grained representation of that chain. In both panels, the cross indicates the center of mass of the polymer.

4.2.2.1 Two-body interaction

In all models, the pairwise interactions were determined using force matching (FM),^{50,139,142} where the force on each CG bead is calculated as the sum of the forces on the monomers of the corresponding polymer. We determined the two-body interaction potentials between the CG polymers using FM from bulk simulations, U_b^{2b} , as well as from simulations containing two isolated polymers, U_0^{2b} . The first approach results in a pair potential that (implicitly) includes many-body effects due to the surrounding polymers, while the second approach provides the pairwise interaction of only two polymer chains.

We introduced a cutoff radius, r_c^{2b} , beyond which the CG particles do not interact with each other. The two-body forces in the CG model, f^{2b} , were parameterized using cubic splines with a uniform grid spacing Δr^{2b} . Thus, there are $K \equiv r_c^{2b}/\Delta r^{2b}$ grid points and basis functions which depend linearly on $2K$ spline coefficients λ_i . Imposing continuity and smoothness conditions, reduces the number of free coefficients to K . These coefficients were then determined by matching the forces in the CG representation to the ones from the MR simulations. To determine the K coefficients, we first divided the MR trajectory in blocks containing each N_s snapshots, and then solved the resulting set of $N_s N_p$ linear equations with a constrained least-squares algorithm for each block. Finally, we tabulated f^{2b} with a grid spacing of $\Delta r_{\text{tab}}^{2b} = 0.01 \sigma$, averaging over the blockwise results. To ensure a smooth decay to zero at r_c^{2b} , we multiplied the tabulated forces with a smoothing function f^{sm} for all distances $r > r^{\text{sm}}$:

$$f^{\text{sm}}(r) = \cos\left(\frac{\pi}{2} \frac{r - r^{\text{sm}}}{r_c^{2b} - r^{\text{sm}}}\right). \quad (4.2.2.1)$$

The CG pair potential U^{2b} was then obtained by numerical integration of f^{2b} . This coarse-graining procedure is implemented in the VOTCA-CSG package.¹¹⁵

The pair potential U_b^{2b} was determined from the MR melt simulations (see Sec. 4.2.1) using a cutoff distance of $r_c^{2b} = 8 \sigma$ and a grid spacing of $\Delta r^{2b} = 0.1 \sigma$. Each block contained 300 frames, and the final potential was computed by taking the average of the 67 blocks. For computing U_0^{2b} , we performed 64 independent simulations of two isolated polymers in a large simulation box ($L = 60 \sigma$), so that they did not interact with their periodic images. The polymers were initialized at large distances and then approached each other due to the monomer-monomer attraction (see Sec. 4.2.1). By evaluating the trajectories of these multiple runs, there were enough data for capturing the forces at all relevant distances. For the FM procedure for calculating U_0^{2b} , we used $r_c^{2b} = 8 \sigma$ and $\Delta r^{2b} = 0.2 \sigma$, averaging over 14 blocks each containing 750000 frames. In both cases, we chose $r^{\text{sm}} = 7.5 \sigma$.

4.2.2.2 Three-body interaction

Three-body interactions were taken into account using the Stillinger-Weber (SW) potential with flexible angular dependence, as implemented in the VOTCA-CSG package¹⁴⁴

$$U^{\text{SW}}(r_{IJ}, r_{IK}, \theta_{IJK}) = f^{\text{SW}}(\theta_{IJK}) \exp\left(\frac{\eta}{r_{IJ} - r_c^{\text{SW}}}\right) \exp\left(\frac{\eta}{r_{IK} - r_c^{\text{SW}}}\right), \quad (4.2.2.2)$$

with angle θ_{IJK} between the three particles (I being the central one), and angular interaction term $f^{\text{SW}}(\theta_{IJK})$. The cutoff distance r_c^{SW} determines how many triplets are included into the local environment of each CG polymer, and U^{SW} smoothly decays to zero when one of the two inter-particle distances r_{IJ} and r_{IK} reaches r_c^{SW} . The parameter η controls the steepness of this decay with small η corresponding to a steeper transition.

The angular dependence of U^{SW} allows for capturing anisotropic interactions, which occur near the polymer-solvent interfaces in the MR simulations (see Fig. 4.2). The angular interaction term $f^{\text{SW}}(\theta_{IJK})$ was fitted to the residual forces of the MR simulations $\Delta \mathbf{f}$ *via* the FM procedure.¹⁴⁴ We determined $\Delta \mathbf{f}$ acting on each CG polymer chain by subtracting the CG two-body force from the total reference force from the MR simulations, \mathbf{f}_{MR} :

$$\Delta \mathbf{f} = \mathbf{f}_{\text{MR}} - \mathbf{f}^{2\text{b}}. \quad (4.2.2.3)$$

In practice, we determined $\Delta \mathbf{f}$ by recomputing the forces acting on the centers of mass of the polymers in the MR trajectories using the pairwise forces $\mathbf{f}^{2\text{b}}$. For $\mathbf{f}^{2\text{b}}$, we either used the forces due to $U_b^{2\text{b}}$ or $U_0^{2\text{b}}$ (see Sec. 4.2.2.1). To distinguish the resulting three-body potentials, we will refer to them as $U_{\Delta b}^{\text{SW}}$ and $U_{\Delta 0}^{\text{SW}}$, respectively. The subscript “ Δ ” indicates that the SW potential was parameterized using the residual forces $\Delta \mathbf{f}$ [Eq. (4.2.2.3)]. Parameterization of U^{SW} according to $\Delta \mathbf{f}$ ensures the orthogonality of the two-body and three-body terms.¹⁴⁴ The fit parameters then depend on the choice of the pair potential, so that the additional three-body contribution can be regarded as a correction term to either $U_b^{2\text{b}}$ or $U_0^{2\text{b}}$.

All trajectory reruns with pair potentials were carried out with the GROMACS simulation package.¹⁶⁶ All reruns with three-body potentials were carried out with the LAMMPS simulation package¹⁶⁷ with the user pair style `sw/table`¹⁴⁴ as available under

<https://gitlab.mpcdf.mpg.de/votca/lammps>.

We used a cubic spline representation for $f^{\text{SW}}(\theta_{ijk})$ with $K^{\text{SW}} = 31$ grid points and a linear dependence on the $2 K^{\text{SW}}$ spline coefficients. Treating the

two exponential terms in Eq. (4.2.2.2) as prefactors led to a linear set of equations which was solved by a constrained least-squares solver. For the thinnest films with $H \approx 4 R_{g,b}$, we averaged over 200 blocks containing each 200 frames, while for the remaining systems we averaged over 800 blocks each containing 50 frames. We systematically varied the cutoff radius of $U_{\Delta b}^{\text{SW}}$ and $U_{\Delta 0}^{\text{SW}}$ from $r_c^{\text{SW}} = 4 \sigma$ to $r_c^{\text{SW}} = 10 \sigma$, and the parameter η from $\eta = \sigma$ to $\eta = 4 \sigma$.

4.2.2.3 Local density-dependent interaction

Alternatively, many-body interactions were included in our CG simulations by supplementing the two-body pair potential with a local density-dependent potential (LDP). In this representation, each CG polymer carries a cloud, which effectively describes its monomers that were integrated out during the coarse-graining procedure (see Fig. 4.3). For simplicity, we assumed that these clouds can be described by radial weight functions $\omega(r_{IJ})$, which only depend on the distance r_{IJ} between the CG particles I and J . In this work, we constructed the weight function $\omega(r_{IJ})$ by computing the pairwise overlap integral between the average monomer density clouds around the polymers centers of mass, $\rho_{\text{cloud}}(r)$. To determine $\omega(r_{IJ})$, we first computed $\rho_{\text{cloud}}(r)$ from the MR bulk simulations:

$$\rho_{\text{cloud}}(r) = \left\langle \frac{1}{N 4\pi r^2} \sum_{i=1}^N \delta(|\mathbf{r}_i - \mathbf{r}_{\text{com}}| - r) \right\rangle, \quad (4.2.2.4)$$

where \mathbf{r}_{com} is the center of mass position of the polymer. Assuming a pairwise overlap of the density clouds, the weight function $\omega(r_{IJ})$ is then given by:

$$\omega(r_{IJ}) = \int_V \rho_{\text{cloud}}(|\mathbf{r}_{\text{com}}^I - \mathbf{r}|) \rho_{\text{cloud}}(|\mathbf{r}_{\text{com}}^J - \mathbf{r}|) d^3\mathbf{r}. \quad (4.2.2.5)$$

Thus, $\omega(r_{IJ})$ can be interpreted as the number density of monomer pairs between a pair of (CG) polymers at center-to-center distance r_{IJ} . Figure 4.4 shows $\rho_{\text{cloud}}(r)$ as well as $\omega(r)$, which are both bounded and monotonically decrease with increasing r . The monomer density decays almost completely at $r \approx R_{g,b}$ while $\omega(r)$ vanishes almost entirely at $r \approx 2 R_{g,b}$, as expected.

The potential energy of CG particle I due to the LDP is then:

$$U_I^{\text{LDP}} = G(\varphi_I) \quad (4.2.2.6)$$

where G is an embedding function, and φ_I is the local density of (fictitious) monomer pairs at the position of CG particle I due to the other CG particles.

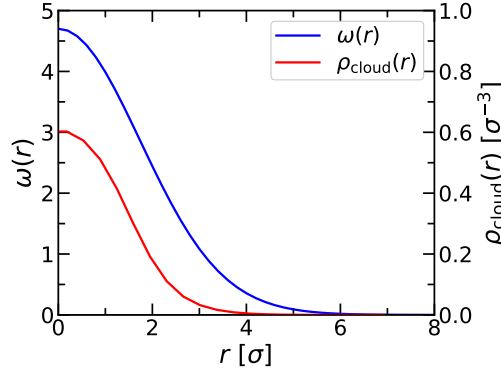


Figure 4.4: Weight function $\omega(r)$ (left axis) and monomer density around the polymer center of mass $\rho_{\text{cloud}}(r)$ (right axis) as functions of radial distance r .

We posit that φ_I can be expressed as the linear superposition of the pair density clouds of the other particles, that is:

$$\varphi_I = \sum_{J \neq I} \omega(r_{IJ}). \quad (4.2.2.7)$$

In this context, the embedding function G is a mean-field representation of many-body effects,¹⁶⁸ and local density gradients enter the force calculation *via* the gradient of the embedding function. Thus, many-body effects are related to the curvature of G : If G is a linear function of φ_I , then the total potential reduces to a two-body pair potential. If G has a positive curvature, *i.e.*, $d^2G/d\varphi^2 > 0$, then the LDP becomes more repulsive as the local density increases. Due to the assumed pairwise additivity of the weight functions $\omega(r_{IJ})$, the LDP can be interpreted as a generalized pair potential, which is why in practice LDPs are typically implemented as pair potentials.^{113,169,170} Hence, the pressure tensor can be calculated as usual *via* the standard Clausius virial equation.⁴⁰

The embedding function was implemented as a cardinal B-spline function, with equally spaced nodes λ_i at a distance of $\Delta\varphi = 0.5$. This representation has the advantage that the derivatives with respect to the amplitude of the nodes, $\partial U^{\text{LDP}}/\partial\lambda_i$, are linear, which facilitates the calculation in the update step [see Eq. (4.2.2.11)]. The number of nodes was adapted during the coarse-graining procedure to cover the required density range. We determined the embedding function G from our MR simulations using relative entropy minimization (REM).^{36,37} The goal of this technique is to optimize the CG potential in such a way that the difference between the probability distribution of the MR configurations in the CG and the MR representation is minimized. This difference can be measured by the relative entropy, also known as the Kullback-Leibler

divergence,⁵³ which can be interpreted as the difference in information conveyed by the probability distributions. So in contrast to other coarse-graining methods like FM or IBI, which are designed to reproduce the potential of mean force or the radial distribution function, respectively, REM aims to minimize the difference in the configurational probability distributions. As a consequence, the resulting CG potentials will differ, because there is in general no unique parameterization which optimizes *all* quantities of the target system.

In the canonical ensemble, the relative entropy can be written as:

$$S_{\text{rel}} = \beta \langle U_{\text{CG}} - U_{\text{MR}} \rangle_{\text{MR}} - \beta (A_{\text{CG}} - A_{\text{MR}}) + \langle S_{\text{map}} \rangle_{\text{MR}} \quad (4.2.2.8)$$

where U_{MR} and U_{CG} denote the potential functions, and A_{MR} and A_{CG} the configurational parts of the Helmholtz free energy for the MR and CG system, respectively. The term $\langle S_{\text{map}} \rangle_{\text{MR}}$ is the (unavoidable) contribution to the relative entropy due to the mapping. The relative entropy S_{rel} can be optimized using approximate and/or iterative methods to find a (locally) optimal set of parameters of the potential. In this work, we used a Newton-Raphson update rule to find the root of $\partial S_{\text{rel}}/\partial \lambda$, that is

$$\lambda^{j+1} = \lambda^j - \left(\frac{\partial S_{\text{rel}}}{\partial \lambda} \right) / \left(\frac{\partial^2 S_{\text{rel}}}{\partial \lambda^2} \right) \quad (4.2.2.9)$$

where j denotes the iteration. Here, the parameters to be optimized are the amplitudes of the nodes λ_i of the cardinal B-spline representation of the embedding function G . To improve the stability of the updates and the convergence of the optimization, we multiplied the change of the parameter $\Delta \lambda_i$ with a constant factor $\alpha = 0.01$ and clipped it at $\pm 1k_{\text{B}}T$. The resulting update rule for the coefficients is then:

$$\lambda_i^{j+1} = \lambda_i^j - \Delta \lambda_i^j \quad (4.2.2.10)$$

$$\Delta \lambda_i^j = \min \left(-1, \max \left(1, \alpha \left[\left\langle \frac{\partial U^{\text{LDP}}}{\partial \lambda_i} \right\rangle_{\text{MR}} - \left\langle \frac{\partial U^{\text{LDP}}}{\partial \lambda_i} \right\rangle_{\text{CG}} \right] \right. \right. \\ \left. \left. \left[\left\langle \frac{\partial^2 U^{\text{LDP}}}{\partial (\lambda_i)^2} \right\rangle_{\text{MR}} - \left\langle \frac{\partial^2 U^{\text{LDP}}}{\partial (\lambda_i)^2} \right\rangle_{\text{CG}} + \beta \left\langle \left(\frac{\partial U^{\text{LDP}}}{\partial \lambda_i} \right)^2 \right\rangle_{\text{CG}} - \beta \left\langle \frac{\partial U^{\text{LDP}}}{\partial \lambda_i} \right\rangle_{\text{CG}}^2 \right]^{-1} \right) \right) \quad (4.2.2.11)$$

where $\langle \cdot \rangle_{\text{MR}}$ and $\langle \cdot \rangle_{\text{CG}}$ indicate averaging in the mapped MR and CG representation, respectively. For all investigated systems, we used 500 iterations with 200000 simulation timesteps per iteration. All simulations with LDPs were carried out on graphics processing units with the HOOMD-blue simulation package (v. 2.4.2).^{113,169,170}

4.3 Results

4.3.1 Two-body interactions

In a bulk polymer melt, every monomer is isotropically surrounded by other monomers, both from the same chain as from the other ones. Consequently, the intra- and inter-chain interactions compensate each other, and the polymers behave as dispersed in a Θ -solvent.^{17,72,1711} This does, however, *not* mean that the effective polymer-polymer interactions vanish, but that they are purely entropic instead. Figure 4.5 shows the effective pair potential between CG polymers in a melt, U_b^{2b} , which is repulsive and bounded, allowing a partial overlap of the CG polymers. This behavior is due to the fractal and open nature of the polymer coils, which lets the centers of mass of two coils be at the same place while each chain can fluctuate without intersecting the other [see Fig. 4.3(a)].

Previous consideration within renormalization-group theory¹⁷⁵ predicted that the effective potential can be approximated rather accurately by a Gaussian function $U_b^{2b} \propto \exp[-(r_{IJ}/\sigma_p)^2]$, with characteristic length-scale of the interaction σ_p . Our simulation data can be fitted perfectly to this functional form with $\sigma_p = 3.4\sigma$. Such Gaussian polymer-polymer interactions have also been reported in previous lattice¹⁷⁶ and off-lattice^{123,175} simulations of athermal polymer solutions. In particular, Louis *et al.* varied the polymer concentration over a wide range, from dilute solutions up to almost five times the overlap concentration, finding that U_b^{2b} barely changed.¹²³ Further, they showed that U_b^{2b} reproduces rather accurately the equation of state of a polymer solution. However, a CG description of the polymers *only* in terms of U_b^{2b} is insufficient for modeling polymer films and droplets, as the CG polymers will repel each other rather than form a stable film, unless additional external constraints are applied.

The pair potential between two isolated polymers, U_0^{2b} , is attractive (see Fig. 4.5), resembling a soft square-well potential with a well depth of about $-13 k_B T$ and a well width of roughly 2σ . This strong polymer-polymer attraction is due to the attractive tail of the LJ interaction between the monomers in the MR model [see Eq. (4.2.1.1)]. At such low polymer concentrations, the employed MR model effectively describes polymers in a poor (implicit) solvent,^{163,177,178} which have collapsed into compact globules with a distinctly smaller radius of gyration compared to their coil-like analogs in the melt, *i.e.*, $R_{g,0} \approx 1.5\sigma$ *vs.* $R_{g,b} \approx 2.2\sigma$. Once two (or more) of these globules approach each other in the MR simulations, they coalesce into a small polymer droplet that remains stable

¹The Flory ideality theorem should be considered with care, however, as there are subtle differences in the static^{172,173} and dynamic¹⁷⁴ properties of ideal and real polymers.

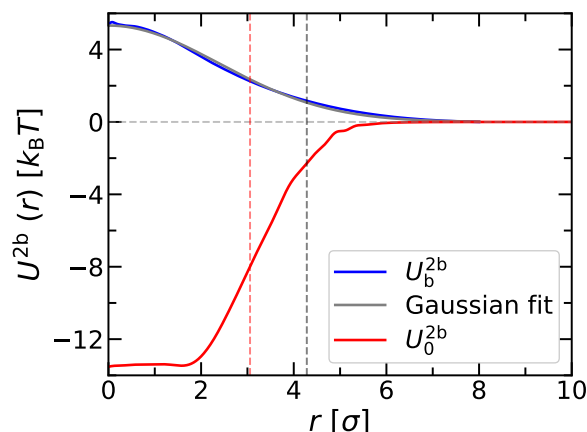


Figure 4.5: Pair potential acting between two coarse-grained polymers, computed from bulk melt simulations (U_b^{2b} , blue) and from simulations containing two isolated polymers (U_0^{2b} , red). The corresponding diameters of the polymers, $2 R_g$, from the microscopically resolved simulations are indicated by a black and red vertical line, respectively.

over the investigated simulation time. However, simulations containing many CG polymers interacting only through U_0^{2b} behave in a nonphysical manner: Due to the strong attraction for $r \lesssim 5 \sigma$ and the lack of repulsive excluded volume interactions at short distances, the CG polymers merge into a single small aggregate, with a diameter that even *decreases* with increasing aggregation number. Alternatively, one could try to parameterize the pair potential at the Θ -temperature, where the solvent quality is exactly poor enough to cancel the polymer-swelling due to excluded volume effects, so that the polymer scales like an ideal chain. However, the resulting pair potential between two isolated polymers will be zero (apart from statistical fluctuations), leading to incorrect thermodynamic properties in the bulk and confined systems.

4.3.2 Three-body SW interactions

We determined the SW three-body potentials for the three different film systems, as well as for the droplet with radius $R \approx 6 R_{g,b}$. For all investigated systems, the forces of the reference systems, f_{MR} , were reproduced most faithfully when the residual forces Δf were fitted using U_0^{2b} with a cutoff distance of $r_c^{SW} = 10 \sigma$. The dependence on η was weak and we chose $\eta = \sigma$ (see the supporting information for a comparison of all different parameter combinations, as well as the results when fitting to the residual forces using U_b^{2b}).

Figure 4.6(a) shows the average force on the CG polymers perpendicular to

the film surface, $F_z(z)$, in the film with $H \approx 8 R_{g,b}$, where measurements were taken every 100τ . (The results for the other systems are qualitatively similar and are included in the supporting information.) In the MR simulations, the average net force on the center of mass of the polymers vanishes in the central bulk-like region of the film, $|z| < 6 \sigma$ due to the film's symmetry along the z -axis. Further away from the film center, the net force on the polymers points inwards with its magnitude increasing the further the chain is located away from the film center. This net attraction reflects the cohesive forces between the monomers in the MR simulations.

Using the particle positions from the MR trajectories (see Sec. 4.2.2.2 for details), we computed the force profiles in the CG model and show the resulting profiles also in Fig. 4.6(a). The resulting net force has a similar shape as the reference force, although the flat plateau in the film center is replaced by a weak oscillation. The contribution from the two-body potential U_0^{2b} leads to a strong net attraction between the CG polymers, which is especially pronounced near the film surfaces because of the inhomogeneous polymer density distribution (see Fig. 4.2). The pairwise attraction in the CG model extends deep into the bulk region of the film, which is (partially) compensated by the repulsive three-body SW interactions. It should be noted, that these force profiles have been created using the CG model in trajectories from the MR reference simulations, and *not* from dedicated simulations using the CG model. Thus, the good agreement of the force profiles shown in Fig. 4.6(a) does not guarantee the stability of the films and droplets in the CG representation, which we will test in Sec. 4.3.4.

In Fig. 4.6(b), we plot the angular part of the SW potential $f^{SW}(\theta_{IJK})$ for the three different films as well as for the droplet with radius $R \approx 6 R_{g,b}$. The fitted functions $f^{SW}(\theta_{IJK})$ have similar shapes in all investigated cases, indicating a good transferability of the three-body potential. As the two exponential terms of the SW potential are strictly positive [see Eq. (4.2.2.2)], a positive (negative) sign of $f^{SW}(\theta)$ results in a net repulsion (attraction) between the central CG particle I of a triplet configuration and the two CG particles J and K . The SW potential is purely repulsive for small angles $\theta_{IJK} < 30^\circ$, where the three CG polymers I , J , and K are lined up with J at a smaller distance and K at a larger distance behind chain J [see Fig. 4.6(c)]. For larger angles $\theta_{IJK} > 30^\circ$, $f^{SW}(\theta)$ oscillates around zero with a rather small amplitude.

4.3.3 Local density-dependent potentials

Using U_0^{2b} as the two-body potential between the CG polymers, we determined U_0^{LDP} for all films and droplets *via* REM, as described in Sec. 4.2.2.3. To verify that the coarse-graining procedure converged, we computed the probability

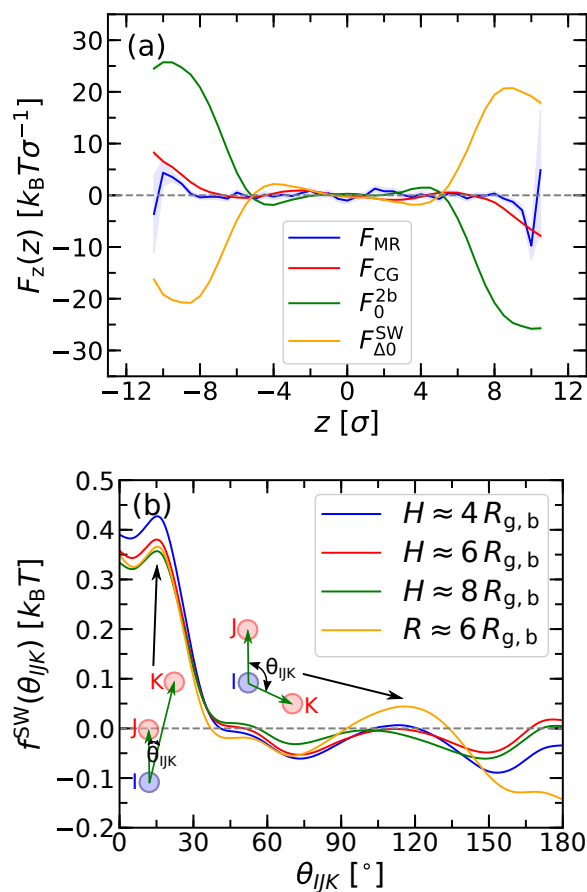


Figure 4.6: (a) Forces acting on the center of mass of the polymers perpendicular to the film surface, $F_z(z)$, as functions of z . Data shown for a film with thickness $H \approx 8 R_{g,b}$. The blue curve corresponds to the net force in the MR simulation, while the red curve shows the net force using the CG potential. The green and orange curves show the two- and three-body contributions, U_0^{2b} and $U_{\Delta 0}^{SW}$, respectively. The shaded regions correspond to the measurement uncertainty determined from the standard error of the mean. (b) Angular part $f^{SW}(\theta_{IJK})$ of $U_{\Delta 0}^{SW}$ for films with different thicknesses H as indicated and droplet with radius $R \approx 6 R_{g,b}$.

density function of the local density, $P(\varphi)$, in both the mapped MR reference simulations as well as in the CG simulations. Figure 4.7(a) shows the corresponding results for the film with thickness $H \approx 8 R_{g,b}$ and the droplet with radius $R \approx 4 R_{g,b}$. The data for the CG and MR simulations are in perfect agreement, which indicates that the LDPs have been parameterized accurately. Small φ values correspond to polymers located near the polymer-solvent interface, while large φ values are associated to polymers in the central bulk-like region of the systems. All $P(\varphi)$ have a maximum near $\varphi \approx 15$, which indicates that there are more polymers in the bulk-like region than at the surfaces. This maximum becomes more pronounced as the film thickness H (droplet radius R) increases, because the surface-to-volume ratio decreases as $2/H$ ($3/R$). Note also, that a droplet with diameter $2R = H$ has a larger surface-to-volume ratio than a film with the same thickness, which is reflected by the larger $P(\varphi)$ values at small φ for droplets compared to films [see Fig. 4.7(a)].

Figure 4.7(b) shows the fitted embedding functions $G(\varphi)$ for all CG systems, which look nearly identical except for a slight vertical shift in the region $\varphi \gtrsim 4$. This good agreement indicates that the obtained LDPs are transferable across the different systems, that is, the embedding function $G(\varphi)$ parameterized for a small droplet is also applicable to a thick film. $G(\varphi)$ increases linearly with φ for $5 \leq \varphi \leq 22$, which corresponds to the range of local densities typically observed in the films and droplets [see Fig. 4.7(a)]. For such linear embedding functions $G(\varphi) = k\varphi + C$, the potential energy of particle I due to the LDP can be written as:

$$U_I^{\text{LDP}} = C + k\varphi_I = C + \sum_{I \neq J} k\omega(r_{IJ}) = C + \sum_{I \neq J} U^{\text{LDP},2b}(r_{IJ}). \quad (4.3.3.1)$$

Hence, in this case, the LDP behaves like an additive pair potential $U^{\text{LDP},2b}(r)$, which restores the excluded volume interactions between the CG polymers. Figure 4.7(c) shows the resulting effective pair interaction between CG particles in the bulk-like region of the films and droplets, $U_0^{2b} + U^{\text{LDP},2b}$, which is repulsive at short distances to prevent the CG particles from collapsing onto each other, and has a minimum at $r \approx R_{g,b} = 2.2\sigma$ to achieve the desired inter-particle spacing in the bulk-like region. Outside the density region $5 \leq \varphi \leq 22$, the derivative $d^2G/d\varphi^2$ is nonzero [see inset of Fig. 4.7(b)], so that many-body effects become relevant. For smaller $\varphi < 5$, the derivative $d^2G/d\varphi^2$ is negative, so that U_0^{LDP} becomes less repulsive as the local density increases. In contrast, $d^2G/d\varphi^2 > 0$ for $\varphi > 22$, which ensures the incompressibility of the polymer film.

To assess the computational performance of the CG model, we performed bulk simulations with $N_p = 2000$ polymers in a cubic box with edge length $L = 35.15\sigma$. Here, we found that the CG simulations using only the two-body

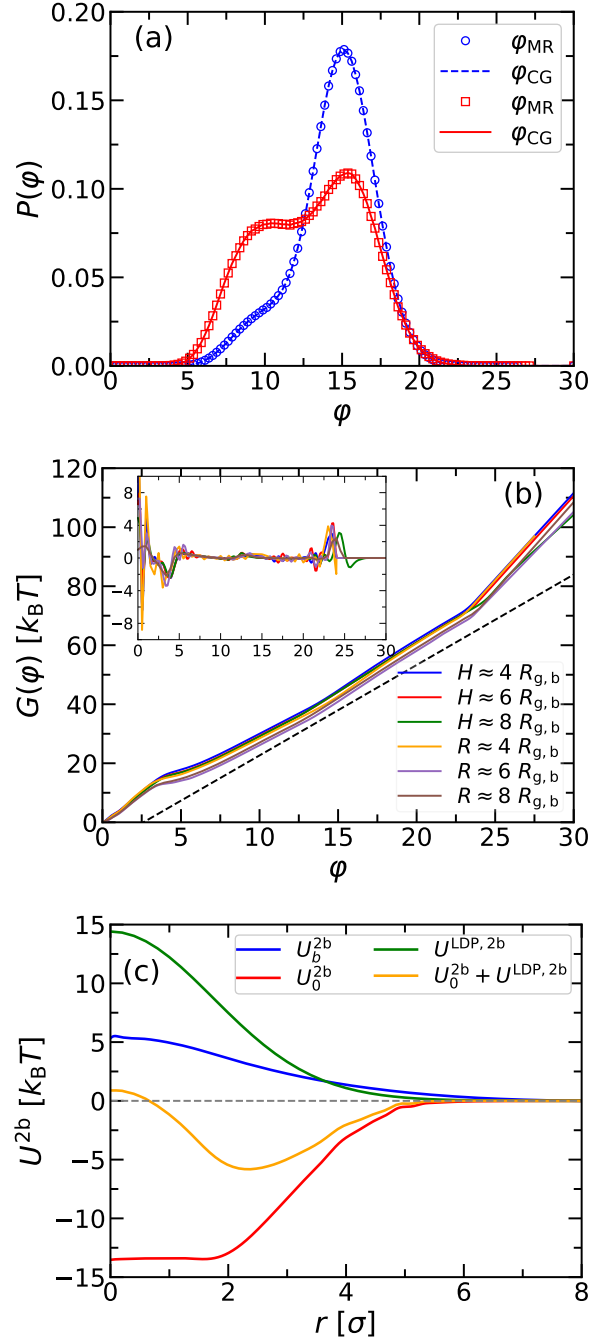


Figure 4.7: (a) Probability density function of the local density, $P(\phi)$, in the MR (symbols) and CG systems (lines). Results for thin films ($H \approx 8 R_{g,b}$) are shown in blue, while results for droplets ($R \approx 4 R_{g,b}$) are shown in red. (b) Embedding functions, $G(\phi)$, for all investigated systems. The dashed line indicates $G \propto \phi$, while the inset shows the second derivative $d^2G/d\phi^2$. (c) Two-body interactions parameterized in the bulk simulations, U_b^{2b} , and effective two-body potential for the CG simulations with an LDP in the bulk region, $U_0^{2b} + U^{LDP, 2b}$. 113

interactions U_b^{2b} achieved about 40 times more timesteps per second than the MR simulations, while the CG simulations with both U_0^{2b} and U_0^{LDP} achieved about 10 times more timesteps per second than the MR simulations. The actual speedup with respect to the physical timescales is likely even higher due to the accelerated dynamics in the soft CG models.

4.3.4 Stability of Simulations

To investigate whether the CG representations using the SW potential or the LDP are able to reproduce stable films and droplets, we performed MD simulations using the CG models starting from the final snapshots of the MR reference simulations. Figure 4.8 shows typical snapshots of the CG simulations of the films with thickness $H \approx 8 R_{g,b}$ at different simulation times. The starting configuration is shown in Fig. 4.8(a), while the results from the runs with the SW potential and LDP are shown in Figs. 4.8(b,c) and 4.8(d,e), respectively.

The films in the CG simulations with U_0^{2b} and $U_{\Delta 0}^{\text{SW}}$ start to become unstable already after about 2τ , as the CG polymers collapse into small spherical aggregates, which are stabilized by a long range repulsion between them. The inability of this combination of two- and three-body interactions to reproduce the MR simulations even qualitatively stems likely from the restricted functional form of $U_{\Delta 0}^{\text{SW}}$. The residual force fit of the SW potential (see Sec. 4.2.2.2) most probably only captures a local minimum of the potential of mean force, whereas the global minimum of the fitted CG model corresponds to this collapsed state. This transition can be better understood by considering the strong attraction of U_0^{2b} at short distances (see Fig. 4.5), whereas the three-body contribution $U_{\Delta 0}^{\text{SW}}$ is repulsive only for small angles $\theta_{IJK} < 30^\circ$, where three CG beads are lined up as shown in Fig. 4.6(b). Thus, small deviations in θ_{IJK} are already sufficient to overcome the repulsive barrier between neighboring particles. This interpretation is corroborated by the distribution of angles θ_{IJK} in the collapsed droplets, where we find almost no triplets with angles in the range of $\theta_{IJK} < 30^\circ$. CG simulations with U_b^{2b} and $U_{\Delta b}^{\text{SW}}$ do not reproduce stable films or droplets either, as in these cases the (attractive) three-body contribution $U_{\Delta b}^{\text{SW}}$ is not sufficient to compensate the repulsive pair potential U_b^{2b} (see the supporting information for the fitting results of $U_{\Delta b}^{\text{SW}}$). In addition, including SW interactions with a relatively large cutoff ($r_c^{\text{SW}} = 10\sigma$) significantly slows down the CG simulations due to the large number of triplets included into the force calculation. In fact, the CG simulations became less efficient than the original MR ones.

In contrast, the CG simulations with the pair potential U_0^{2b} and LDP U_0^{LDP} lead to stable film and droplet configurations, even after long simulation times (see Fig. 4.8). (Note that simulations using the effective pair interactions U_0^{2b}

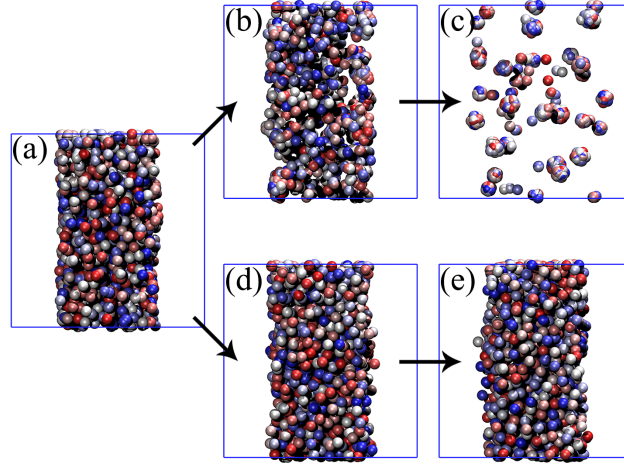


Figure 4.8: Snapshots of the CG simulations of the films with thickness $H \approx 8 R_{g,b}$. (a) Starting configuration. (b,c) CG simulations with U_0^{2b} and $U_{\Delta 0}^{SW}$ at $t = 2\tau$ and 4τ , respectively. (d,e) CG simulations with U_0^{2b} and U_0^{LDP} at $t = 2\tau$ and 40000τ , respectively.

and $U^{LDP,2b}$, see Sec. 4.3.3, exhibit a similar instability as the simulations with U_0^{2b} and $U_{\Delta 0}^{SW}$.) To investigate the resulting configurations in more detail, we computed the density profiles along the z -axis for the films and in the radial direction for the droplets. The corresponding results for the film with thickness $H \approx 8 R_{g,b}$ and the droplet with radius $R \approx 4 R_{g,b}$ are plotted in Fig. 4.9. Again the measurement uncertainties of the density profiles are estimated from the standard error of the mean between ten subdivided blocks of the data. Overall, the density profiles in the CG simulations are reasonably close to the ones from the MR simulations, with relative deviations of H and R below 4%, and similar polymer densities in the central region of the films and droplets. However, the polymer-solvent interfaces are slightly broader in the CG simulations, which is also reflected by the smaller surface tension measured in the CG simulations, that is $\gamma = 0.31 \pm 0.01 k_B T / \sigma^2$ vs. $\gamma = 1.42 k_B T / \sigma^2$ for the planar films. Further, the peaks in the density profiles near the polymer-solvent interfaces are slightly more pronounced and more narrow in the MR simulations compared to the CG simulations. This difference can be understood if one considers the oblate ellipsoidal shape of the polymers at the interfaces in the MR simulations [see Fig. 4.2(b)]. This shape anisotropy near the polymer-solvent interface can, however, not be captured by our CG model, where the polymers are modeled as soft spheres.

In addition, we computed the force profiles along the z -axis in the films and in the radial direction r for the droplets (see Fig. 4.10). The force profiles were

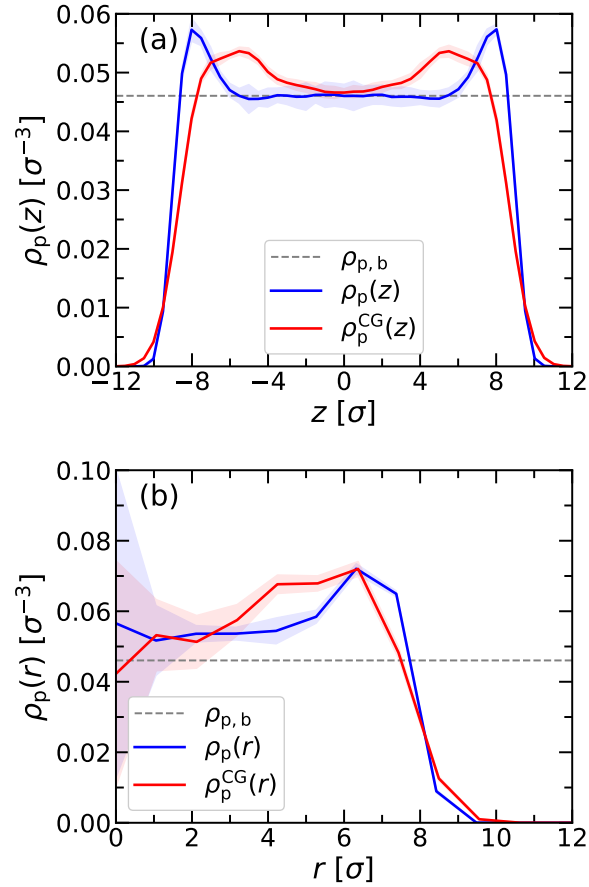


Figure 4.9: Number density profiles of polymer centers of mass of the mapped MR reference (ρ_p) and CG simulations (ρ_p^{CG}) for (a) a film with thickness $H \approx 8 R_{g,b}$, and (b) a droplet with radius $R \approx 4 R_{g,b}$. The horizontal dashed line indicates the polymer number density in the bulk system. In both panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

calculated as the mean forces of particles at the given z or r position with data taken every 100τ . In both the MR and CG simulations, the net force is essentially zero in the central region of the film and droplet, due to the symmetry of the systems. Polymers near the surface of the film/droplet experience attractive forces toward the center of the film/droplet. These cohesive forces are stronger and act on a much narrower region in the MR simulations compared to the CG simulations. If one decomposes the net force into the contributions from the two- and many-body interactions, one sees that U_0^{2b} and U_0^{LDP} have opposite signs and are each significantly larger than the resulting net force. Both forces also extend much further into the central region of the films and droplets than expected from the width of the bumps at the extremities of the force profiles. Evidently, the stability of the films and droplets is the result of a delicate balance between two- and many-body interactions in our CG model.

4.4 Conclusions

We developed coarse-grained (CG) models of homopolymers, where each polymer was represented by a single (soft) spherical particle and the solvent was included implicitly. We focused on simulations of thin films and droplets with strong density inhomogeneities near the polymer-solvent interfaces. In CG simulations where the polymers interacted only through two-body potentials, the films and droplets either dissolved or collapsed into small aggregates, depending on whether the effective polymer-polymer interactions were parameterized in reference simulations in the bulk or at infinite dilution. These CG representations failed at capturing (even qualitatively) the main physical characteristics of the reference systems, because they did not correctly reproduce the cohesive forces and the compressibility of the polymers, respectively.

To address these inherent issues, we supplemented the two-body potentials by additional three-body or many-body interactions, which were parameterized in the inhomogeneous systems. The three-body interactions were represented by a Stillinger-Weber potential which was fitted to the residual forces of the microscopically resolved reference simulations. Many-body interactions were included in a mean-field way *via* local density-dependent interactions, which were optimized using the relative entropy minimization technique. The CG models with three-body interactions failed at maintaining stable polymer films or droplets, and the CG particles collapsed into small spherical aggregates instead. In contrast, the CG simulations with local density-dependent interactions reproduced stable films and droplets with linear dimensions close to the reference simulations, except that the polymer-solvent interfaces were slightly sharper and the accompanied surface tension was higher in the reference sim-

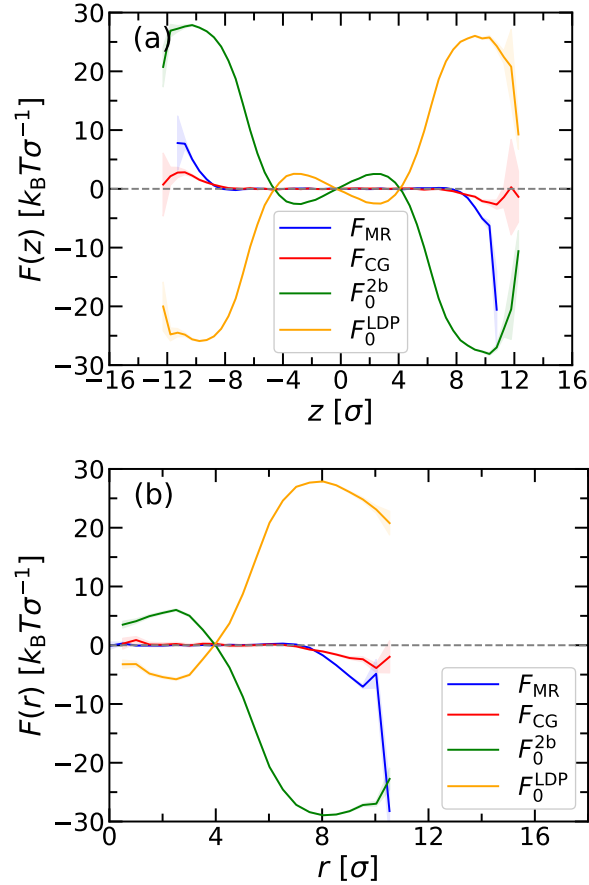


Figure 4.10: Force profiles for (a) a film with thickness $H \approx 8 R_{g,b}$ and (b) a droplet with radius $R \approx 4 R_{g,b}$ as functions of z or r , respectively. The blue curve corresponds to the net force in the MR simulation, while the red curve shows the net force in the CG simulations. The green and orange curves show the two-body and LDP contributions, U_0^{2b} and U_0^{LDP} , respectively. In both panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

ulations. We attributed these differences to the deformation of polymers near the interfaces, which could not be resolved in the CG representation, where the polymers were mapped to spherical beads. In future work, we plan to enhance our model to capture such effects. Further, we want to develop analytic expressions for the pair- and many-body interactions to avoid (or at least minimize) running microscopically resolved reference simulations for parameterizing the CG model.

Acknowledgments

This work was funded by the German Research Foundation (DFG) through project number 233630050 - TRR 146. AN further acknowledges financial support provided by the DFG through project number NI 1487/2-1 and NI 1487/2-2. Computing time was granted on the supercomputer Mogon at Johannes Gutenberg University Mainz (www.hpc.uni-mainz.de).

CHAPTER 5 --- Limitations of the Ultra-Coarse-Grained Model

In this chapter, we will apply the parametrization scheme used in Ch. 4 to chemically more specific systems and discuss its generalization to longer polymer chains. This chapter is structured as follows: In Sec. 5.1 we will present the reference simulations and methods used in this chapter. In Sec. 5.2 we show, how some of the properties needed for our coarse-graining procedure can be obtained from analytical expressions, which can reduce the amount of costly reference simulations. In Sec. 5.3 we will analyze the transferability of our ultra-coarse-graining procedure as well as its limitations and the reasons thereof. Finally, in Sec. 5.4 we will conclude and give an outlook about possible applications and mitigation strategies to improve our coarse-graining procedure.

5.1 Reference Simulations and Coarse-Graining Procedure

5.1.1 Microscopically Resolved Reference Systems

In this chapter we use two different models to which our coarse-graining procedure is applied.

One of these is the same Kremer-Grest (KG) model as discussed in Sec. 4.2.1 with different numbers of monomers per polymer, $N = 40$ and $N = 100$ instead of $N = 20$. As the parametrization of the KG model is the same as the one used previously, we will not repeat the specifics and refer to the corresponding section for a detailed description.

The other model is a bead-spring model of polyethylene (PE) with the force field chosen according to a modified *TraPPE-united-atom* (TraPPE-UA) model of alkanes.¹⁷⁹ In this parametrization the chain consists mostly of beads combining (CH_2), with beads of (CH_3) located at both ends of the chain. The non-bonded interactions between all particles in the system are given by a Lennard-Jones potential, generalized to different types of atoms and smoothed

at the cutoff via a smoothing function $S(r_{ij})$,

$$U^{\text{LJ}}(r_{ij}) = \begin{cases} 4\varepsilon_{t_i t_j} \left[\left(\frac{\sigma_{t_i t_j}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{t_i t_j}}{r_{ij}} \right)^6 \right] S(r_{ij}), & r_{ij} < r_c \\ 0, & r_{ij} \geq r_c \end{cases} \quad (5.1.1.1)$$

where $t_i, t_j \in \{A, B\}$ denotes the type of particle i and j respectively and the cutoff is chosen as $r_c = 14\text{\AA}$. The smoothing function $S(r_{ij})$ is given by

$$S(r_{ij}) = \begin{cases} 1, & r_{ij} < r_{\text{on}} \\ \frac{(r_c^2 - r_{ij}^2)^2 (r_c^2 + 2r_{ij} - 3r_{\text{on}}^2)}{(r_c^2 - r_{\text{on}}^2)^3}, & r_{\text{on}} \leq r_{ij} \leq r_c \\ 0, & r_{ij} \geq r_c \end{cases} \quad (5.1.1.2)$$

with the onset radius $r_{\text{on}} = 13\text{\AA}$. This modification to the original TraPPE-UA model ensures that both the LJ potential as well as its derivative, *i.e.* the force, vanish at the cutoff.

The corresponding strength and length parameters are summarized in Tab. 5.1. The inter-species parameters were calculated using the *Lorentz-Berthelot rules*,^{180,181}

$$\varepsilon_{t_i t_j} = \sqrt{\varepsilon_{t_i t_i} \varepsilon_{t_j t_j}} \quad (5.1.1.3)$$

$$\sigma_{t_i t_j} = \frac{\sigma_{t_i t_i} + \sigma_{t_j t_j}}{2}. \quad (5.1.1.4)$$

| Bead | Atoms | LJ strength ε/k_B [K] | LJ size σ [\AA] | Mass m [u] |
|------|--------------------|-----------------------------------|-----------------------------------|------------------|
| A | (CH ₃) | $\varepsilon_{AA} = 98$ | $\sigma_{AA} = 3.75$ | $m_A = 15.0345$ |
| B | (CH ₂) | $\varepsilon_{BB} = 46$ | $\sigma_{BB} = 3.95$ | $m_B = 14.0266$ |

Table 5.1: Parameters for non-bonded LJ-interaction in PE model. The inter-species parameters were computed using the Lorentz-Berthelot rules, *cf.* Eqs. (5.1.1.3) and (5.1.1.4). The unit of mass, u , denotes the atomic mass unit.

The polymer bonds, which in the original TraPPE-UA model are modeled as stiff bonds, are implemented using harmonic potentials between neighboring particles in our case,

$$U^{\text{bond}}(r_{ij}) = \frac{1}{2}k(r_{ij} - r_0)^2 \quad (5.1.1.5)$$

with the bond rest length $r_0 = 1.54\text{\AA}$ chosen like in the original model. The spring constant was chosen relatively high as $k = 10^5 k_B T \text{\AA}^{-2}$ so that the bonds

are very strong, mimicking the perfectly stiff bonds.

Furthermore, the model incorporates angle and dihedral potentials via a harmonic angle potential and an *optimized potential for liquid simulation* (OPLS)¹⁸² dihedral force:

$$U^{\text{ang}}(\theta_{ijk}) = \frac{1}{2}k_a (\theta_{ijk} - \theta_0)^2 \quad (5.1.1.6)$$

$$U^{\text{dih}}(\phi_{ijkl}) = \frac{1}{2}C_1 (1 + \cos(\phi_{ijkl})) + \frac{1}{2}C_2 (1 - \cos(2\phi_{ijkl})) \\ + \frac{1}{2}C_3 (1 + \cos(3\phi_{ijkl})) \quad (5.1.1.7)$$

where θ_{ijk} denotes the angle of neighboring particles i , j and k and ϕ_{ijkl} denotes the dihedral angle of neighboring particles i , j , k and l . The angle potential strength is given as $k_a = 62500k_B\text{Krad}^{-2}$ and the rest angle as $\theta_0 = 114^\circ$. The coefficients in the OPLS potential described in Eq. (5.1.1.7) are set to $C_1 = 355.03 k_B\text{K}$, $C_2 = -68.19 k_B\text{K}$ and $C_3 = 791.32 k_B\text{K}$ in accordance with the TraPPE-UA model.

As we already established the transferability of our approach between droplets and thin films of different radii and thicknesses, we focused on thin films exclusively in this part of the analysis of the coarse-grained model.

All MD simulations were conducted in the canonical \mathcal{NVT} ensemble unless stated otherwise explicitly. In both cases, the number of particles is given by the number of polymers N_p times the number of monomers N per polymer, $\mathcal{N} = N_p N$, with $N \in \{40, 100\}$ for the KG model and $N = 200$ for the PE model. In the case of the KG model, the simulation parameters were chosen consistent with the simulations in Ch. 4, with the only difference being the aforementioned degree of polymerization, *i.e.* the number of monomers per chain, and the number of polymers in the corresponding simulation boxes.

For the simulations of PE on the other hand we chose a temperature of $T = 800\text{K}$, which is rather high in order to have a higher chance of a non-vanishing density in the gas phase, using a Langevin thermostat with a friction coefficient of $\xi = 1\frac{m}{\tau}$ with a mass scale of $m = 1u$, where u denotes the *atomic mass unit*. For the length scale we chose $\sigma = 1\text{\AA}$ and for the energy scale we chose $\varepsilon = 1k_B T$ such that the derived unit of time is $\tau = \sigma\sqrt{m/\varepsilon} \approx 38.77\text{fs}$. The timestep of the Velocity Verlet scheme used to integrate the equations of motion was set to $\Delta t = 0.003\tau$. In the case of the bulk systems, we had $N_p = 1473$ polymers for the KG model with $N = 40$, $N_p = 589$ polymers for the one with

$N = 100$, and $N_p = 400$ polymers for the PE model. The simulations of thin films consisted of $N_p = 656$, $N_p = 724$, and $N_p = 400$ polymers respectively.

While all simulations of the KG model were conducted in cubic boxes as before, the simulations of the PE model used cubic boxes for the bulk simulation and the simulation at infinite dilution and an elongated rectangular box for the simulation of the film.

As was done previously, we first simulated the bulk polymer melt in the \mathcal{NPT} ensemble at zero pressure for the sake of equilibration, followed by a \mathcal{NVT} simulation to obtain the equilibrium properties in this ensemble.

The creation of the thin film proceeded analogously to the description in Sec. 4.2.1, so we refer to this section for more detailed information.

Figure 5.1 illustrates the monomer densities $\rho(z)$ and the polymer COM densities $\rho_p(z)$ for all systems. Furthermore, in both cases, the bulk density ρ_b is given as a reference. We continue using the convention from Ch. 4, where a subscript “ b ” stands for properties obtained from bulk simulations and subscript “ 0 ” denotes those obtained at infinite solution.

As one can see in this picture, the density profiles of the reference simulations of the KG model qualitatively match those from Ch. 4 with the same excess regions of ρ_p^{CG} at the surface and the same monomer density in the center of the films. Besides that, the monomer density obtained from the bulk simulations matches the previous results as well, as is expected. The density profiles for PE do not exhibit such distinct peaks, with the excess at the surface less pronounced compared to the KG model. The bulk monomer density of $\rho_b = 0.0355 \text{ nm}^{-3}$ corresponds to a specific mass of $\rho_b \hat{=} 0.82745 \text{ g/cm}^3$ which is slightly below typical specific masses of 0.857 g/cm^3 to 0.967 g/cm^3 ,¹⁸³ which can be attributed to the high temperature of 800 K.

The density profiles of the coarse-grained systems however, differ significantly from the reference and show an almost even distribution of CG beads in the box. We will discuss the reason for this in more detail in Sec. 5.3.

5.1.2 Coarse-Graining Procedure

In this analysis, we did not change the coarse-graining procedure for the systems interacting via a two-body interaction and a local density-dependent interaction but omitted the coarse-graining procedure with interactions comprised of a two-body potential and a three-body Stillinger-Weber potential, as we already established how and why this approach fails in Sec. 4.3.2 and Sec. 4.3.4. Hence,

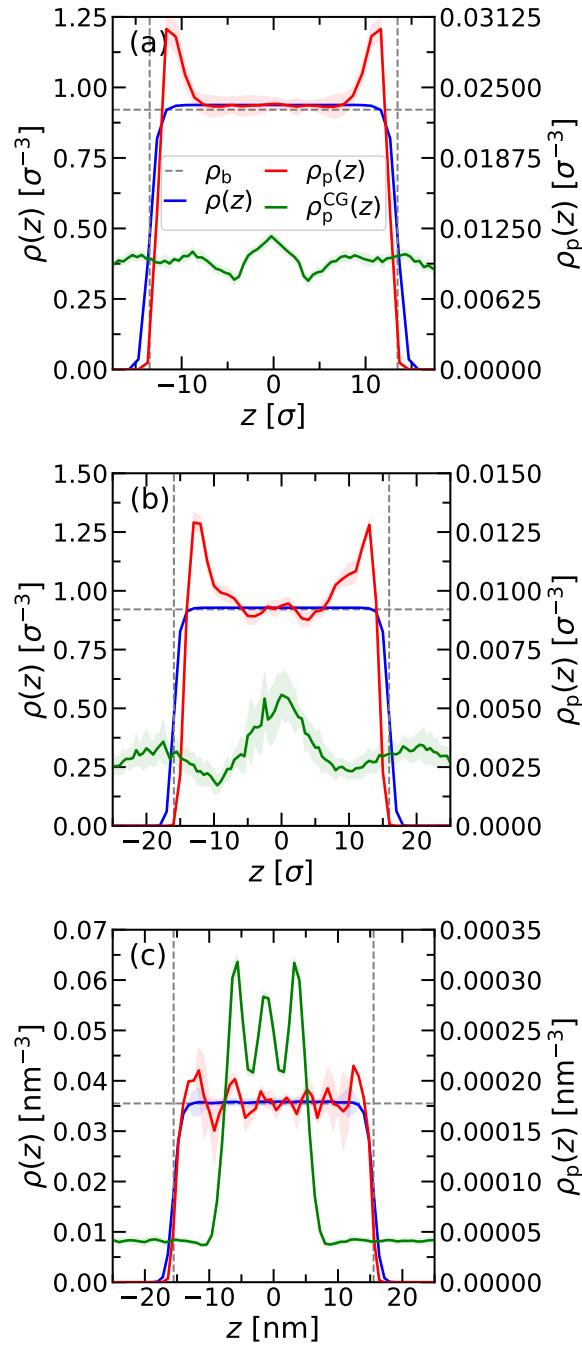


Figure 5.1: Comparison of density profiles for (a) the KG model with $N = 40$, (b) the KG model with $N = 100$ and (c) the polyethylene model with $N = 200$. The legend in panel (a) is valid for panel (b) and panel (c) as well. In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

the coarse-grained beads of the PE polymers were again located at the center of mass, *i.e.* at

$$\mathbf{r}_{\text{com}} = \frac{1}{M} \sum_{i=1}^N m_i \mathbf{r}_i \quad (5.1.2.1)$$

where $M = \sum_{i=1}^N m_i$ denotes the total mass of a polymer.

We parametrized the two-body interaction using force matching at infinite dilution as before. More information on the general procedure for determining the pair potential can be found in Sec.4.2.2. For the KG systems, we used a total of 400 (/800) simulations at infinite dilution and chose the size of the box such that the polymers were far enough away to not interact with their periodic images.

For the local density-dependent potential, we continued using the relative entropy minimization to parametrize the embedding function G , which is still represented using cardinal B-splines. The spacing of the splines was adapted to the systems in such a way, that the local density distribution is covered by the same number of nodes as in Ch. 4.

The definition of the local density was again chosen as before, see Eqs. (4.2.2.4) and (4.2.2.5). The corresponding results are depicted and discussed in Sec. 5.2, where we also compare the results with analytical considerations.

5.2 Analytical Expressions for Reference Properties

As the microscopically resolved reference simulations are very costly, it is desirable to reduce the number of these simulations and obtain the properties necessary for our procedure from analytical expressions. While the simulations at infinite dilution are computationally not very intense, the simulations of thin films and the bulk simulations are computationally expensive.

Nevertheless, as can be seen from Fig. 4.7(a), the probability density functions of the local density that are extracted from the simulations of the inhomogeneous systems and needed for the relative entropy minimization strongly depend on the system. Because the local density intrinsically incorporates many-body contributions it is not possible to obtain or approximate these probability density functions without the knowledge of the n -particle distribution functions of higher order. Obtaining these distribution functions, however, would require atomistic simulations again, which defeats the purpose of the current considerations. Nonetheless, as the highest values of the local density

are obtained under bulk conditions, which are universal for all systems of the same model provided the systems are large enough, all corresponding probability density functions of the local density show a peak at the same value; in Fig. 4.7 this peak is located at about $\varphi = 15$. By using the bulk radial distribution function of the center of masses $g_b(r)$ one can determine the location of this peak φ_{peak} approximately by computing

$$\varphi_{\text{peak}} = \langle \varphi \rangle_b \quad (5.2.0.1)$$

$$\approx \int_0^\infty (4\pi\rho_b g_b(r)) r^2 \omega(r) dr \quad (5.2.0.2)$$

$$= 4\pi\rho_b \int_0^\infty g_b(r) r^2 \omega(r) dr \quad (5.2.0.3)$$

where ρ_b is the global density of polymers. The approximation used to arrive at Eq. (5.2.0.3) assumes a low density, such that contributions from higher order correlations can be neglected. The radial distribution function in the bulk can easily be obtained from scaling considerations, so that the location of the local density peak can be approximated quite easily.

The most expensive simulations are the bulk simulations which are conducted to obtain the monomer densities around the centers of mass in particular. This quantity can be obtained from fundamental considerations and has been derived previously by Yamakawa,¹⁸⁴ based on work by Debye and Bueche.¹⁸⁵ They showed that the probability $P(\mathbf{r})$ of finding a segment of an ideal chain consisting of N segments with a bond length of b at a displacement \mathbf{r} from the center of mass is given by

$$P(\mathbf{r}) = \frac{1}{N} \sum_{i=1}^N \left(\frac{3}{2\pi \langle S_i^2 \rangle} \right)^{3/2} \exp \left(-\frac{3|\mathbf{r}|^2}{2 \langle S_i^2 \rangle} \right) \quad (5.2.0.4)$$

where

$$\langle S_i^2 \rangle = \frac{1}{3} N b^2 \left[1 - \frac{3i(N-i)}{N^2} \right]. \quad (5.2.0.5)$$

As discussed in Sec. 2.2.2.2, the segmentation of a chain into Kuhn segments yields a polymer with the statistics of an ideal chain, in particular with regards to the contour length and the radius of gyration. Hence, by inserting the Kuhn length as bond length and the number of Kuhn segments in Eq. 5.2.0.4, this expression is applicable also for non-ideal chains. In principle, one would have

to determine the Kuhn length from simulations, in our application from bulk simulations. However, the Kuhn length has often been studied previously and is therefore a readily available property, in contrast to the n -particle distribution functions of higher order needed for the probability distribution function of the local density.

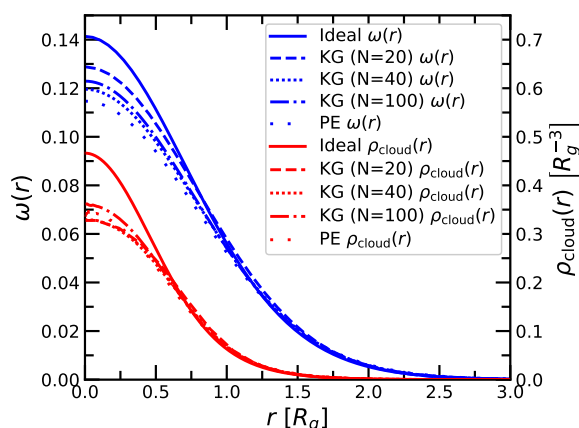


Figure 5.2: Comparison of scaled monomer densities around center of mass and resulting weight functions used in our coarse-graining procedure.

In Fig. 5.2 we compare the scaled distributions of monomers around the polymer center of mass as well as the resulting weight functions for our coarse-graining procedure. As can be seen, the curves coincide for $r > 0.75R_g$ with significant deviations for smaller values of $r < 0.5R_g$. We attribute this deviation to poor statistics for small distances as well as excluded volume effects and chain stiffness. In practice these discrepancies do not have a significant effect as they are very local and can partly be accounted for in the embedding function during the optimization routine.

Therefore, the expression in Eq. (5.2.0.4) can be calculated and scaled to obtain an approximation for the monomer density around the center of mass without conducting any bulk simulations, provided the Kuhn length is known.

5.3 Analysis of Transferability and Limitations of Coarse-Grained Model

In Fig. 5.3 we provide a plot of the potentials resulting from the force matching procedure at infinite dilution described in Secs. 4.2.2.1 and 5.1.2. The x -axis in this plot is rescaled using the corresponding mean value of the radius of gyration

in the bulk to make them comparable. Furthermore, the energy is given in terms of the respective thermal energy. As one can see, the shapes of the potentials are very similar, with the onset of the potentials at approximately the same distance of about $1.7 - 2 R_g$. The depth of the potential on the other hand differs in all plots and does not follow an obvious rule. This is to be expected, as this value depends on both the interaction strength ε and the potential cutoff r_c as well as the degree of polymerization N and the typical inter-molecular distances between those monomers. The latter is not necessarily calculable, as the surface of two polymers close-by is not negligible compared to the volume occupied by them, such that one can not simply use the radial distribution function in the bulk to calculate the potential depth as the mean inter-molecular energy of all monomers of the polymers.

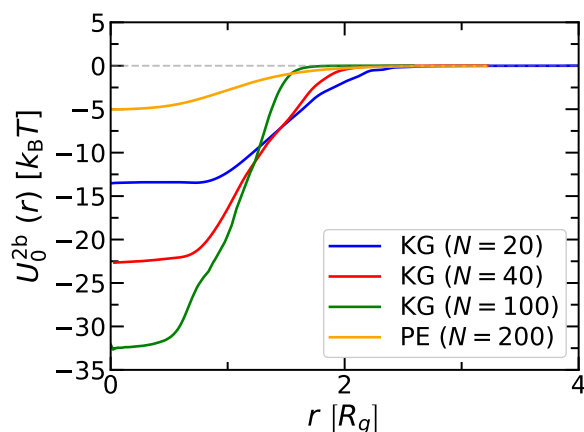


Figure 5.3: Comparison of pair potentials obtained from force matching at infinite dilution for different systems.

When using the vacuum potential obtained for the PE model in our coarse-graining procedure, it turns out that the relative entropy minimization does not give rise to an embedding function that reproduces the local density distribution even after careful tuning. The procedure instead leads to a parametrization that yields a local density distribution with very distinct peaks within the local density region dictated by the reference local density distribution, as can be seen in Fig. 5.4.

This observation indicates, that the coarse-grained system is dominated by many small clusters with different numbers of associated CG beads. This conclusion is reinforced by the fact that the distance between the observed peaks $\langle \Delta\varphi \rangle = 0.32$ is relatively close to the maximum value of the weight function $\max(\omega(r)) = 0.45$ and the peaks are approximately located at multiples of this

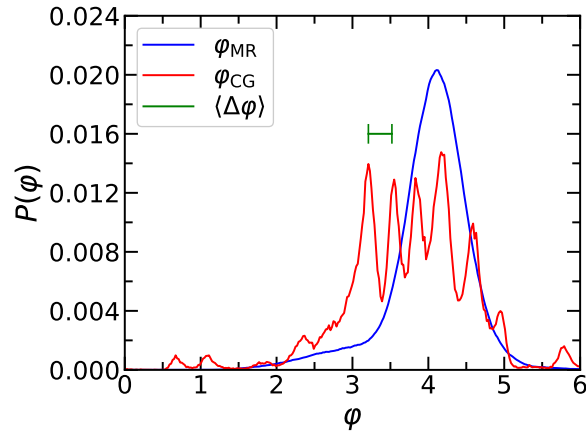


Figure 5.4: Comparison of the reference local density distribution of the polyethylene model and the one obtained via the coarse-graining procedure. The green line indicates the distance between the sharp peaks.

maximum value. While this is not exactly the case, $\langle \Delta\phi \rangle$ is close to values of the $\omega(r)$ for small r , so that the particles do not have to be located strictly at the same position but rather in each other's vicinity. Hence, we deduce that these peaks correspond to isolated clusters consisting of a certain number of CG beads, which is in fact confirmed by a visualization of some simulation snapshots provided in Fig. 5.5, where multiple distinct clusters can be found.

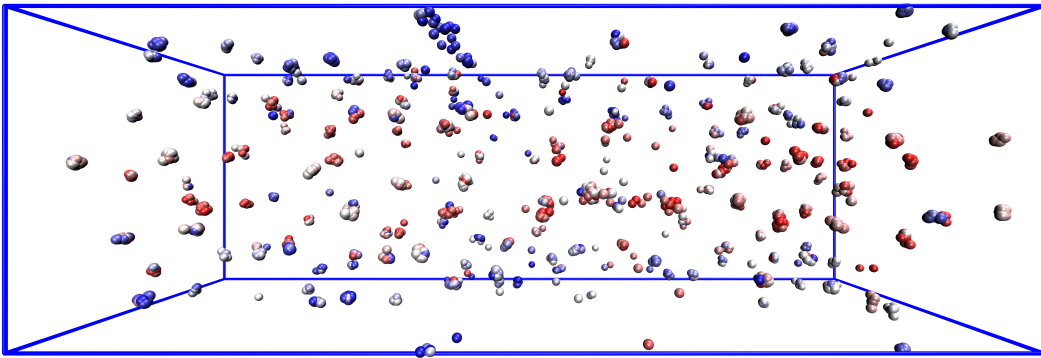


Figure 5.5: Snapshot of coarse-grained polyethylene system. The beads are not drawn to scale of their pair potential to make the clusters more visible. To further increase the distinguishability the beads are colored by their index despite representing identical chemical compositions.

We surmise that this failing of the coarse-graining procedure is due to the size of the polymers which allows for more interpenetration of the polymers and consequently to shorter distances of the centers of mass relative to the radius of gy-

ration of the chains. This leads to an increase of the ratio $R = \varphi_{\text{peak}} / \max(\omega(r))$ of the mean value of the local density under bulk conditions φ_{peak} and the maximum value of the weight function ω . While this value is

$$R_{\text{KG},20} \approx 3.14 \quad (5.3.0.1)$$

$$R_{\text{KG},40} \approx 5.03 \quad (5.3.0.2)$$

$$R_{\text{KG},100} \approx 24.67 \quad (5.3.0.3)$$

for the KG models with different degrees of polymerization it reads

$$R_{\text{PE},200} \approx 9.3 \quad (5.3.0.4)$$

for the PE system. We assume, that for larger values of this ratio it is more favorable for more particles to be close-by, which leads to local optima which the relative entropy minimization is trapped in.

To test this hypothesis, we compared the local density distribution and the density profiles of our coarse-grained KG models consisting of different numbers of monomers per chain with their corresponding reference simulations. While our coarse-grained models are capable of approximately reproducing the local density distribution for both case, see Fig. 5.6, the density profiles in Fig. 5.1 and a visual inspection reveal that the CG beads are more or less evenly distributed in the box and do not form a stable thin film.

The fact that the CG models have the same distribution of the local density as the reference but the resulting density profiles differ significantly from the corresponding references implies, that there is no unique solution to the relative entropy minimization in the above cases, which causes the parametrized model to not form stable films.

Despite the lack of the distinct peaks seen in Fig. 5.4, we see this as a confirmation of our hypothesis and assume that the ratio R is an indicator for how much the polymers interpenetrate. As stated before, this interpenetration leads to a good matching of the local density distribution not implying a matching of the density profiles, as with increasing R there are more and more local optima to be expected.

This is supported by the density profiles in Fig. 5.1 where the CG bead density $\rho_{\text{p}}^{\text{CG}}$ for the KG model with $N = 100$ shows more variance than that with $N = 40$, hinting at smaller local clusters, as well as by the snapshots provided in Fig. 5.7 where one can see more narrower clusters in the case of $N = 100$ and broader ones in the case of $N = 40$.

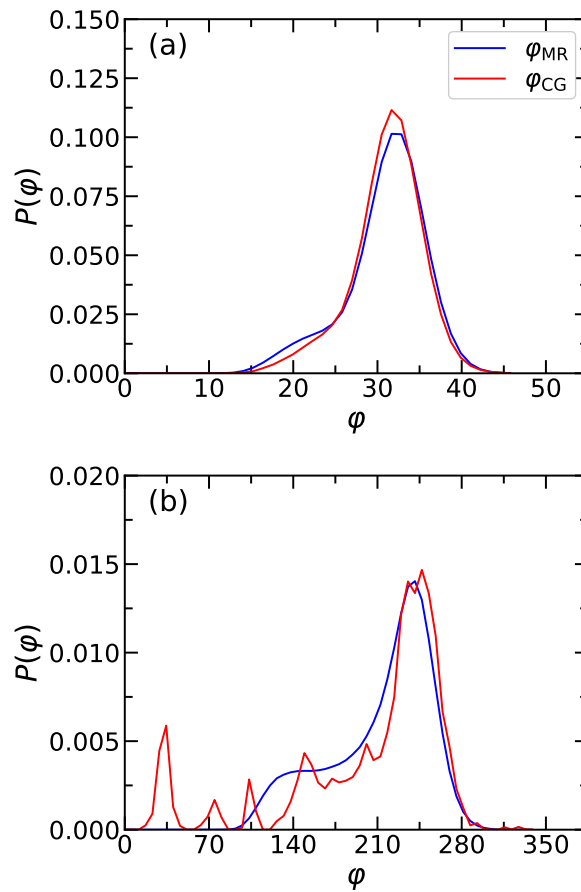


Figure 5.6: Comparison of the reference local density distribution of the KG model for (a) $N = 40$ and (b) $N = 100$ with the corresponding results from the coarse-graining procedure.

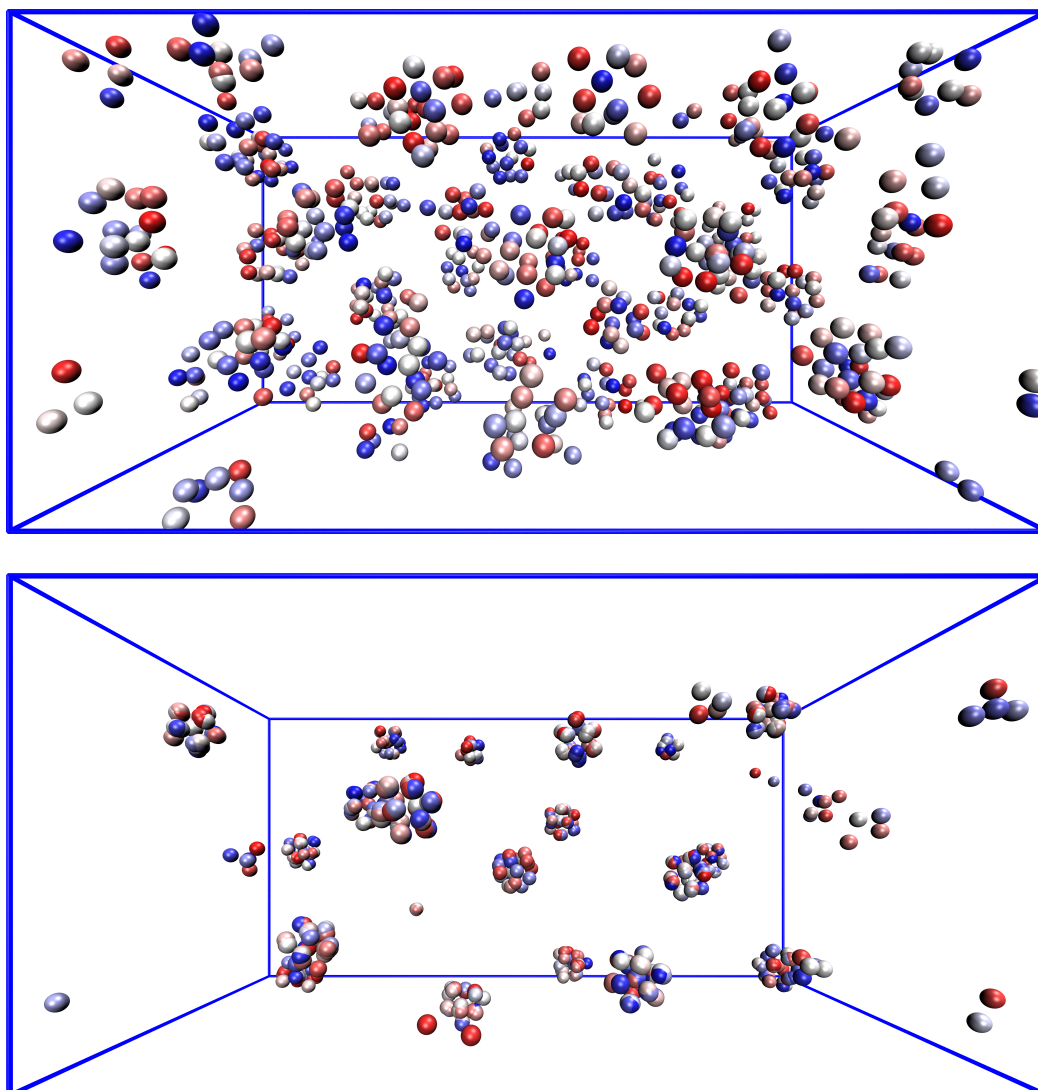


Figure 5.7: Snapshot of coarse-grained KG systems with $N = 40$ (upper panel) and $N = 100$ (lower panel) monomers per chain. The beads are not drawn to scale of their pair potential to make the clusters more visible. To further increase the distinguishability the beads are colored by their index despite representing identical chemical compositions. The clusters formed for $N = 40$ are significantly broader than those for $N = 100$, where the CG beads are overlapping significantly, similar to the case of the PE system.

5.4 Conclusions and Outlook

In this chapter, we investigated the capability of generalization of our ultra-coarse-grained model as well as its limitations. We found that when the polymers in the reference system become too large and the corresponding coarse-grained beads too soft as a consequence, our coarse-graining procedure does not yield the correct structural properties of the reference system even when approximately reproducing the target distribution of the local density.

We surmise that in this case the polymers can interpenetrate more, so that our approach of using a two-body local density leads to local optima of the relative entropy, where the system forms multiple small clusters. This is confirmed by our results for KG chains of length $N = 40$, where the local density distribution is matched quite well by the CG model, but instead of forming very distinct thin films the CG beads exhibit a non-vanishing density throughout the box.

To mitigate this problem, one might introduce a three-body local density-dependent potential - possibly additionally - in order to better resolve and distinguish the local density of many small clusters compared to an almost homogeneous thin film. Furthermore, this would address the previously discussed issues of the anisotropy of the polymers at the surfaces of the inhomogeneous systems, indicated by the excess of polymer centers of mass there.

Despite the aforementioned limitations, it is worthwhile continuing to study our coarse-graining procedure as it might be improved in order to be applicable also to larger polymers. Furthermore it can be applied already to systems with smaller molecules, as the correlation between agreement in the local density distribution and the density profiles seems to increase for smaller systems. Also, one has to bear in mind that by mapping whole chains into single beads one loses a lot of information, which might not be retrievable with such a simple parametrization as described in this thesis. Finally, we want to conclude by stating that the approach of using local density-dependent properties might be extended also to other use cases, *e.g.* to local density-dependent friction coefficients.

CHAPTER 6 --- Conclusion and Outlook

In this thesis we first gave a short introduction to statistical physics, polymer physics, coarse-graining and machine learning followed by a presentation of two contributions to the field of coarse-graining in soft matter research.

The first contribution consisted of forward and inverse design using machine learning methods, in particular neural networks. In the corresponding work, we trained neural networks to learn mappings from pair potentials to the resulting equation of state as well as from the radial distribution function to the effective pair potential leading to it. The first task is interesting when trying to approximately determine the behavior of a certain system at different state points, allowing for rapid prototyping when searching for materials with a desired equation of state. The neural networks solving the second task on the other hand can be used for improving current coarse-graining techniques. Most established coarse-graining techniques are iterative methods, *i.e.* they consist of running multiple simulations, and depend on a good starting point to minimize the number of said simulations required for convergence. Our neural network can in principle be used to provide an accurate first approximation of the effective pair potential, hence reducing the computation time of the established methods applied afterwards.

In both tasks, we focused mostly on the representation of the input and output of the neural network in order to gain insight about the information conveyed by the respective representation. We found that by including our prior knowledge about statistical physics, we were able to improve the performance of our machine learning models significantly, particularly their capabilities for generalization. This is especially important as every single data point used for training stems from expensive molecular dynamics simulations, so that the number of training examples was very limited which made data efficiency imperative. This small number of training examples is typically quite problematic for machine learning methods, as they often rely on large amounts of data.

The second contribution is an ultra-coarse-graining procedure where whole polymer chains are mapped into single beads. As a parametrization using pair potentials exclusively is not sufficient for the coarse-grained system to form stable films or droplets like the reference systems do, we tried supplementing either a three-body Stillinger-Weber potential or a local density-dependent po-

tential. While the three-body potential does not lead to stable films or droplets, the coarse-grained model using the local-density dependent potential is able to do so. Furthermore, we find that applying this procedure to films with different thicknesses or droplets with different radii results in the same parametrization. This indicates that our coarse-graining procedure yields parametrizations that are transferable across different systems consisting of the same polymers with different sizes and shapes of the phases. We further investigated the generalization capabilities of the coarse-graining procedure to different polymers with different degrees of polymerization. As it turns out, there seems to be no unique correspondence between the distribution of the local density, which was optimized in our work, and the density profiles. This leads to coarse-grained representations that reproduce the local density distribution but do not form stable films for larger polymers, which we attribute to their increasing interpenetration. Finally, we pointed out some mitigation strategies to overcome this issue as well as possible extensions of the local density-dependent approach to other physical quantities.

In conclusion, both approaches presented here are pioneering works aimed at making a contribution to statistical physics and coarse-graining. They are both promising in their own regards and are but first steps to improved forward and inverse design as well as more transferable coarse-graining techniques.

CHAPTER **A** BoltzmaNN: Supplementary Information

A.1 Filtering of data

In order to identify simulations that formed heterogeneous structures, the simulation box was divided into 64 subboxes of equal size for every snapshot of the simulation. Then the number density of particles was calculated for each subbox, followed by a calculation of the standard deviation between these subbox densities. Heterogeneous structures could then be identified by a large ratio between the standard deviation and the average density ρ . In cases where this analysis was inconclusive, we visually checked the simulation snapshots.

A.2 Virial Expansion

One typically used extension of the ideal gas law to real gases is the virial expansion

$$\beta P = \rho + \sum_{i=2}^{\infty} B_i(T) \rho^i \quad (\text{A.2.0.1})$$

with virial coefficients $B_i(T)$. For example, the analytical expressions for the second and third virial coefficient of a homogeneous fluid are given by¹¹⁴

$$B_2(T) = -\frac{1}{2} \int_V f(r, T) \, d\mathbf{r} \quad (\text{A.2.0.2})$$

$$B_3(T) = -\frac{1}{3} \int_V \int_V f(r, T) f(r', T) f(|\mathbf{r} - \mathbf{r}'|, T) \, d\mathbf{r} d\mathbf{r}' \quad (\text{A.2.0.3})$$

with the Mayer f -function

$$f(r, T) = \exp[-\beta U(r)] - 1. \quad (\text{A.2.0.4})$$

The second virial coefficient B_2 depends only on the pair interaction between the particles, while B_3 depends on two- and non-additive three-body interactions. In principle, these coefficients can be derived from the grand canonical partition function or from diagrammatic methods.¹¹⁴ In practice, however, it is

rather challenging to determine these coefficients, especially when multi-body interactions are involved. In the low-density limit $\rho \rightarrow 0$, the equation of state is dominated by pairwise terms so that the sum in Eq. (A.2.0.1) can be truncated at $i = 2$. In isotropic systems, one can solve the integral in Eq. (A.2.0.2) in polar coordinates, yielding

$$B_2(T) \approx -2\pi \int_0^\infty f(r, T) r^2 dr. \quad (\text{A.2.0.5})$$

Thus for $\rho \rightarrow 0$, the EOS can be directly computed from the pair potential $U(r)$

$$\beta P \approx \rho - 2\pi\rho^2 \int_0^\infty f(r, T) r^2 dr. \quad (\text{A.2.0.6})$$

To improve the prediction accuracy at higher densities, one can either consider higher order virial coefficients $B_3, B_4, \text{etc.}$ [see Eq. (A.2.0.1)], or introduce an effective second virial coefficient, $B_2^*(T, \rho)$, which implicitly contains the higher order terms in the form of an explicit density dependence

$$\beta P \approx \rho + \rho^2 B_2^*(T, \rho). \quad (\text{A.2.0.7})$$

This term cannot be computed analytically anymore, but it can be readily extracted from computer simulations, *e.g.*, by measuring the pressure P in simulations in the canonical ensemble at various temperatures T and densities ρ .

A.3 Potential prediction from pair distribution function

For a homogeneous and isotropic system, the low-density limit of the radial pair distribution function $g(r)$ is equal to the Boltzmann factor of the pair potential¹¹⁴

$$g_0(r) := \lim_{\rho \rightarrow 0} g(r) = \exp[-\beta U(r)]. \quad (\text{A.3.0.1})$$

Thus, if $g(r)$ is known, one can invert this relation to determine an expression for the pair potential for $\rho \rightarrow 0$

$$U_0(r) := -k_B T \ln[g(r)]. \quad (\text{A.3.0.2})$$

Note, that we replaced $U(r)$ with $U_0(r)$ in Eq. (A.3.0.2), as the resulting potential $U_0(r)$ is *not* necessarily the same as $U(r)$ anymore when $\rho \gg 0$. This procedure

for computing an approximate pair potential is often referred to as Boltzmann inversion, and it can be refined using techniques like RMC⁹¹ or IBI,⁹³ where the interaction potential is iteratively optimized until it reproduces the target radial pair distribution function $g(r)$ with sufficient accuracy. In RMC, an adjustment of the potential is accepted with a probability of

$$p = \min [1, \exp(-\Delta\chi_i^2/2)] \quad (\text{A.3.0.3})$$

where the measure for the agreement between target and calculated $g_i(r)$ of the i -th iteration, χ_i^2 , is given by

$$\chi_i^2 = \sum [g(r) - g_i(r)]^2 / \alpha^2. \quad (\text{A.3.0.4})$$

The parameter α adjusts the sensitivity of χ_i^2 to deviations in the radial distribution functions. In IBI, an adjustment of the potential is done *via*

$$U^{i+1}(r) = U^i(r) - k_B T \ln \left[\frac{g^i(r)}{g(r)} \right] \quad (\text{A.3.0.5})$$

As a starting point, one typically uses $U_0(r)$ from Eq. (A.3.0.2).

Figures A.1 and A.2 show the effective potentials $U^i(r)$ and corresponding radial pair distribution functions $g^i(r)$ that we obtained after performing the IBI procedure on the training and test systems presented in Fig. 3.10(b) and Fig. 3.10(d) of the main manuscript, respectively. To expedite the iterative optimization, we ran the simulations in a smaller box with $L = 10 \sigma$ (instead of $L = 40 \sigma$). We performed 200 iterations of 10^6 timesteps each ($\Delta t = 0.001 \tau$), starting from either the prediction from our NN, $\hat{U}(r)$, or the (effective) pair potential in the low density limit, $U_0(r)$. The mean squared error between the reference radial pair distribution function $g(r)$ and the ones obtained from either $\hat{U}^i(r)$ or $U_0^i(r)$ are shown in Fig. A.3 for the first 25 iterations.

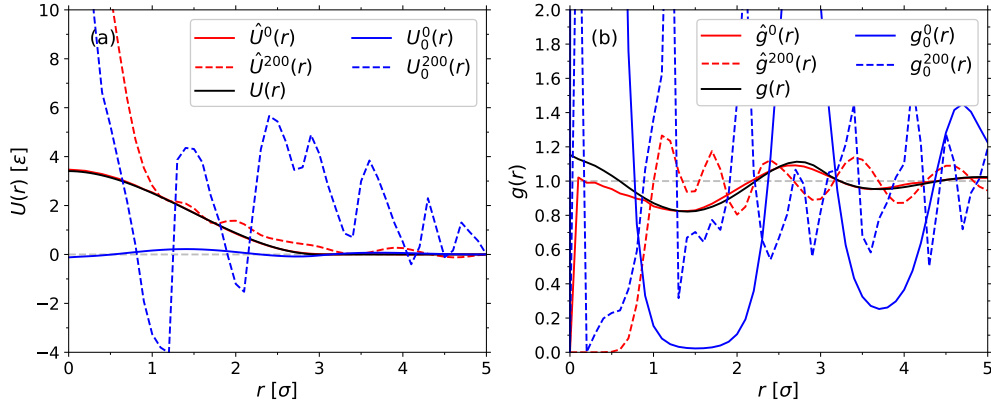


Figure A.1: (a) Comparison between the effective pair potentials for the system shown in Fig. 3.10(b) of the main manuscript. The black solid line shows the target potential $U(r)$, while the red and blue solid lines correspond to the prediction from our NN, $\hat{U}(r)$, and the (effective) pair potential in the low density limit, $U_0(r)$, respectively. The dashed lines show both potentials after 200 IBI steps. (b) Corresponding radial pair distribution functions.

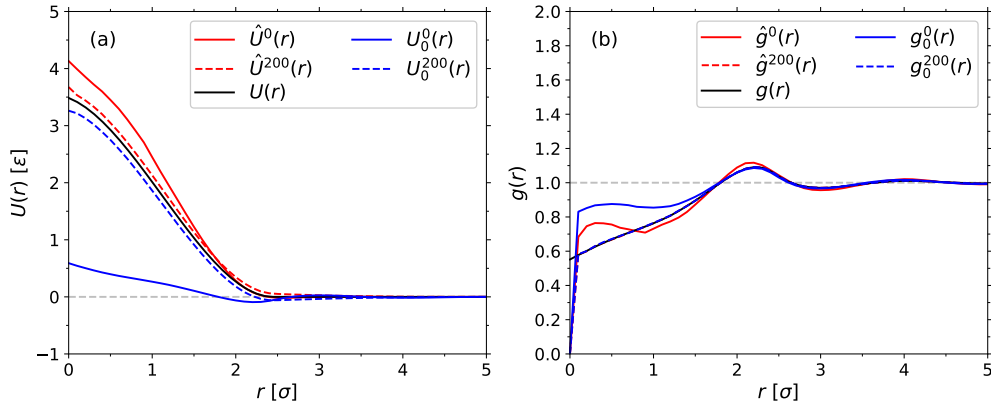


Figure A.2: Same as Fig. A.1, but for the system shown in Fig. 3.10(d) of the main manuscript.

A.3. Potential prediction from pair distribution function

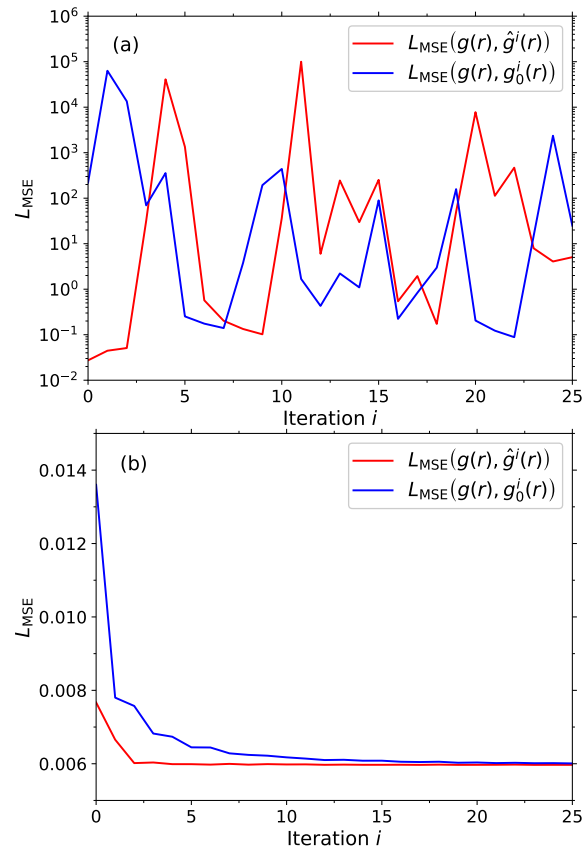


Figure A.3: Mean squared error between the reference radial pair distribution function $g(r)$ and either $\hat{g}^i(r)$ (red line) or $g_0^i(r)$ (blue line). Results shown for the systems shown in (a) Fig. A.1 and (b) Fig. A.2.

A.4 Schematics of network architectures

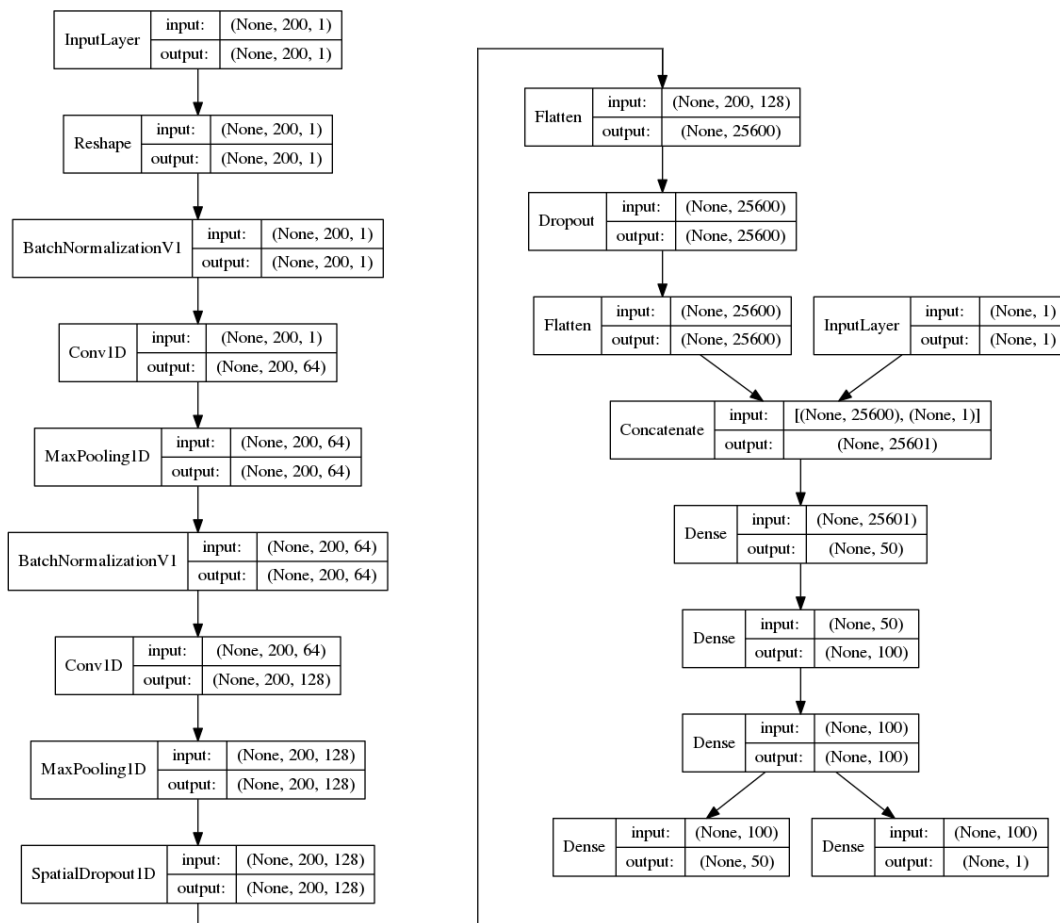


Figure A.4: Schematic representation of the final CN/f network for predicting the (effective) pair potentials.

A.4. Schematics of network architectures

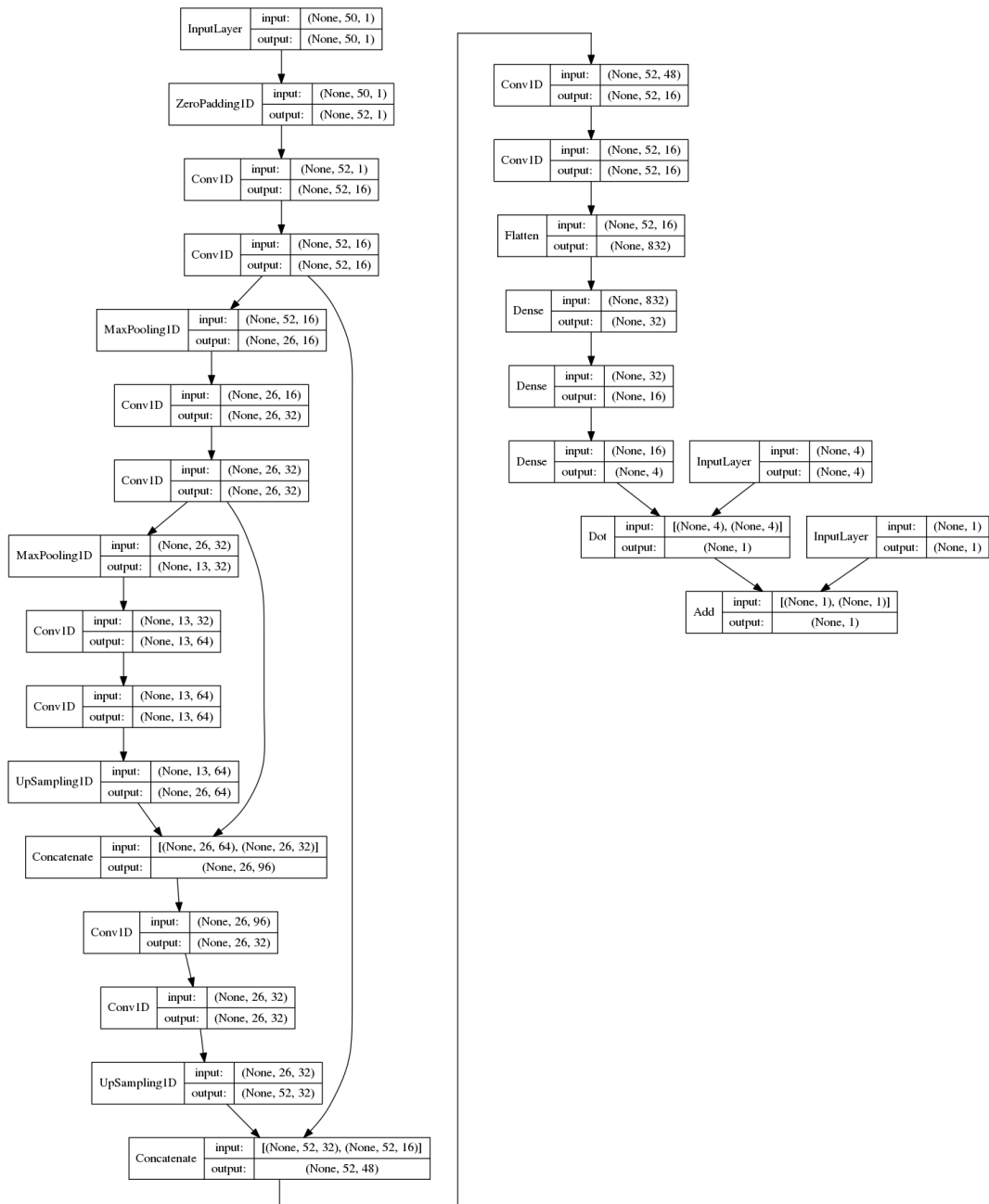


Figure A.5: Schematic representation of the final UN/c network for predicting the pressure.

CHAPTER **B**

**Ultra-Coarse-Graining:
Supplementary
Information**

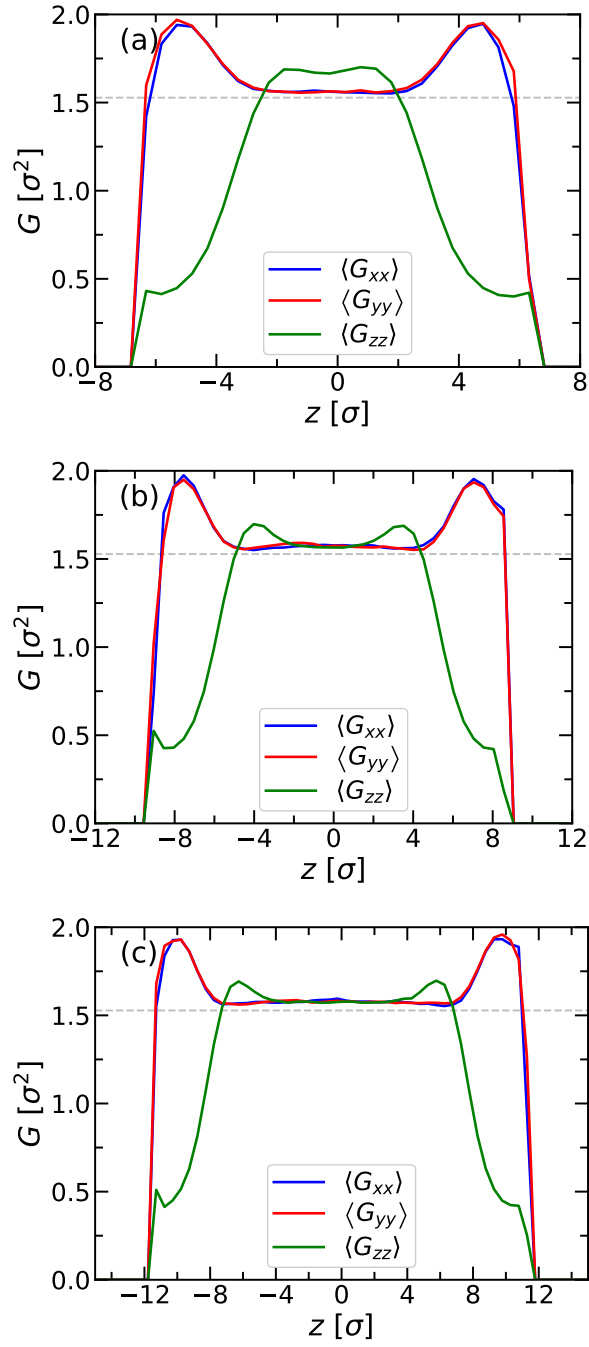


Figure B.1: Diagonal components of the gyration tensor as functions of z for a film with thickness (a) $H \approx 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$.

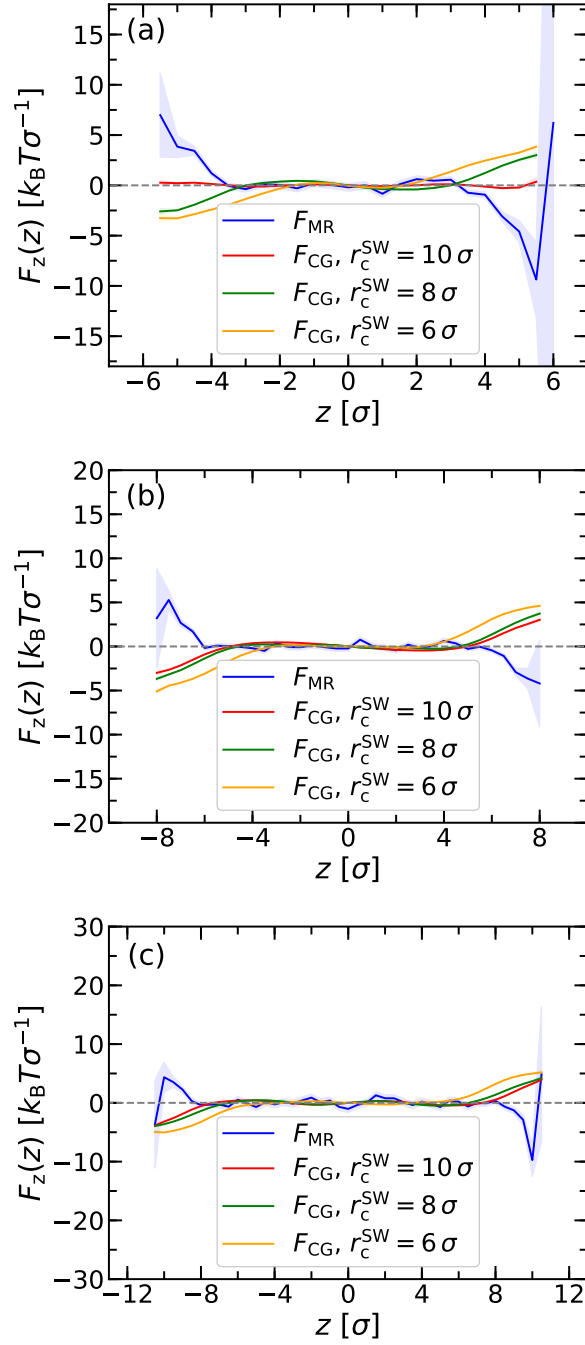


Figure B.2: Force profiles along the z -axis as functions of z for a film with thickness (a) $H \approx 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$. Residual force parameterization with pair potential U_b^{2b} at $\eta = \sigma$ and different cutoffs r_c^{SW} , as indicated. In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

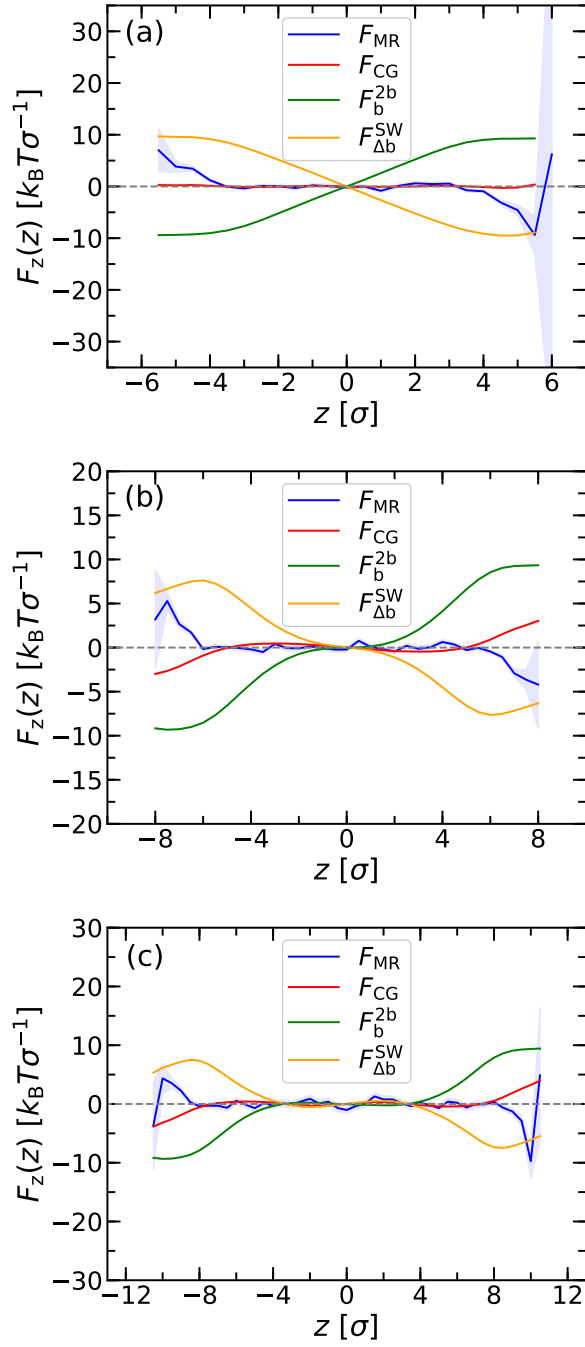


Figure B.3: Force profiles along the z -axis as functions of z for a film with thickness (a) $H = 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$. Residual force parameterization with pair potential U_b^{2b} at $r_c^{SW} = 10.0 \sigma$ and $\eta = \sigma$. In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

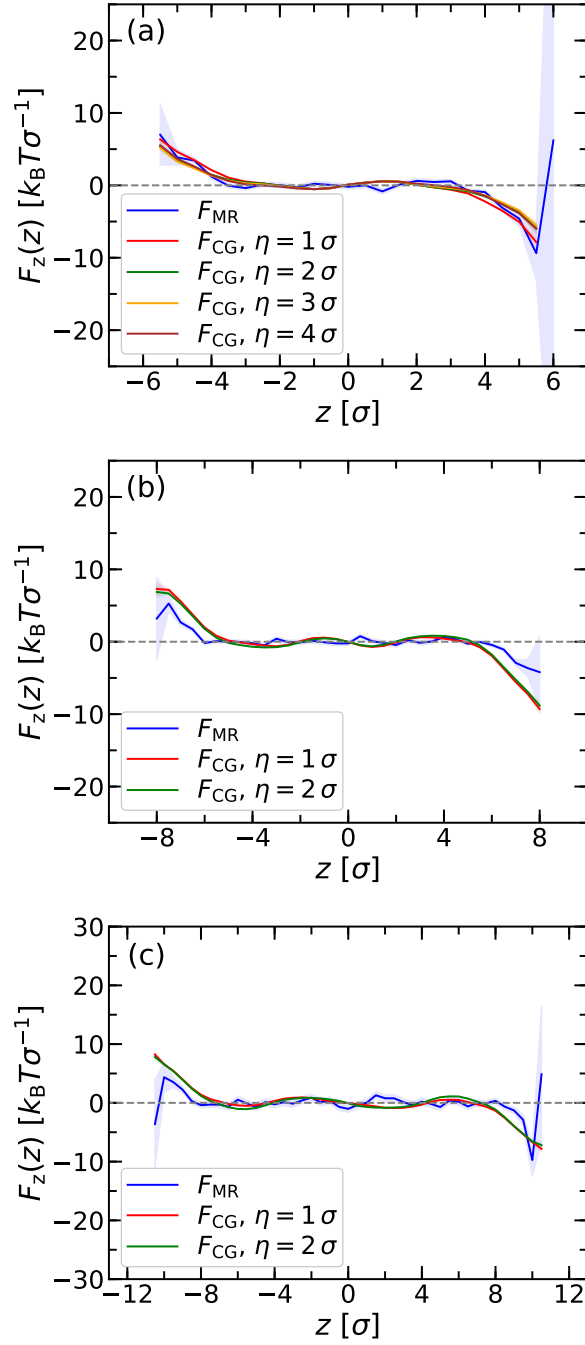


Figure B.4: Force profiles along the z -axis as functions of z for a film with thickness (a) $H = 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$. Residual force parameterization with pair potential U_0^{2b} at $r_c^{SW} = 10.0 \sigma$ and different parameters η , as indicated. In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

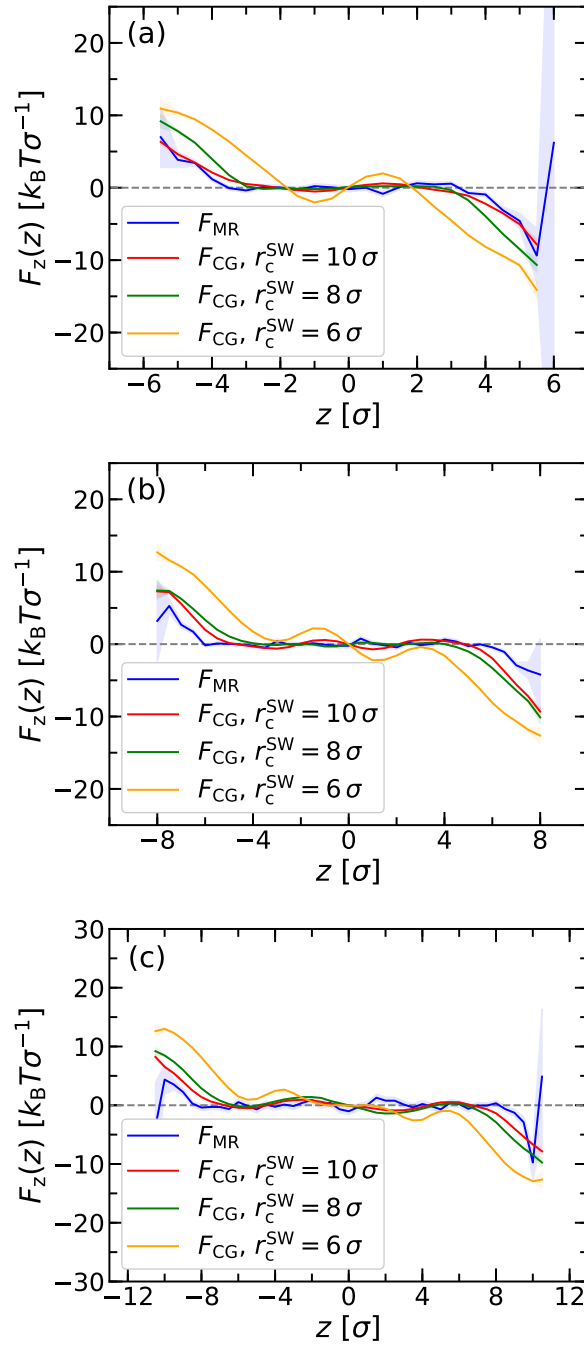


Figure B.5: Force profiles along the z -axis as functions of z for a film with thickness (a) $H = 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$. Residual force parameterization with pair potential U_0^{2b} at $\eta = \sigma$ and different cutoffs r_c^{SW} , as indicated.

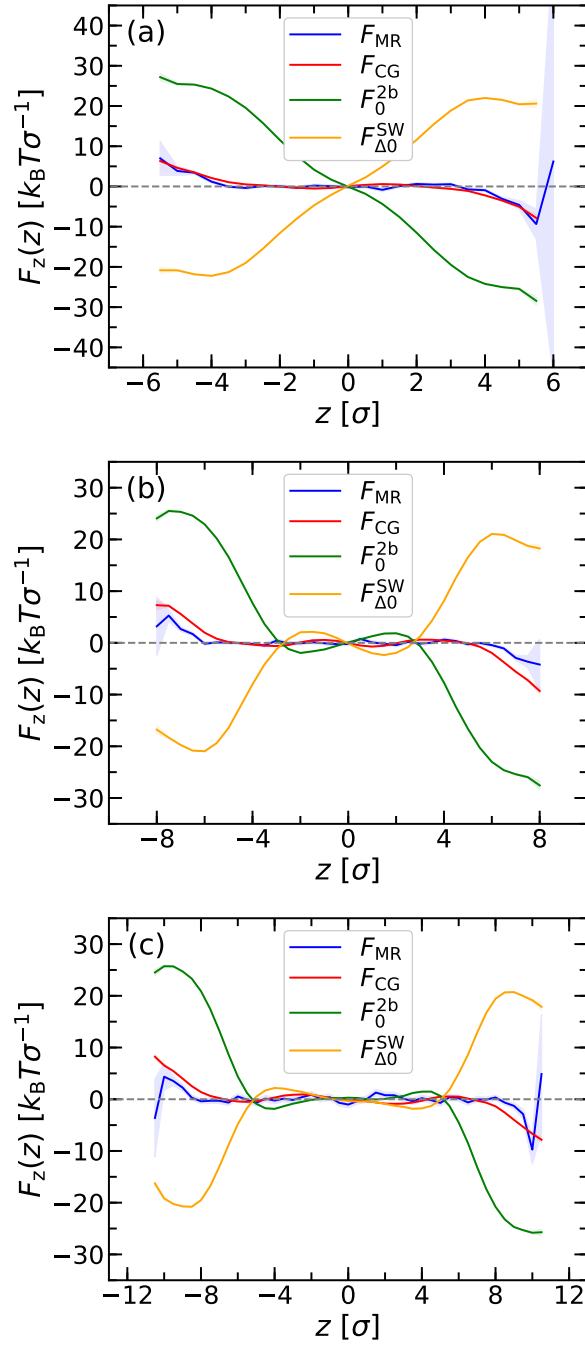


Figure B.6: Force profiles along the z -axis as functions of z for a film with thickness (a) $H = 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$. Residual force parameterization with pair potential U_0^{2b} at $r_c^{SW} = 10.0 \sigma$ and $\eta = \sigma$. In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

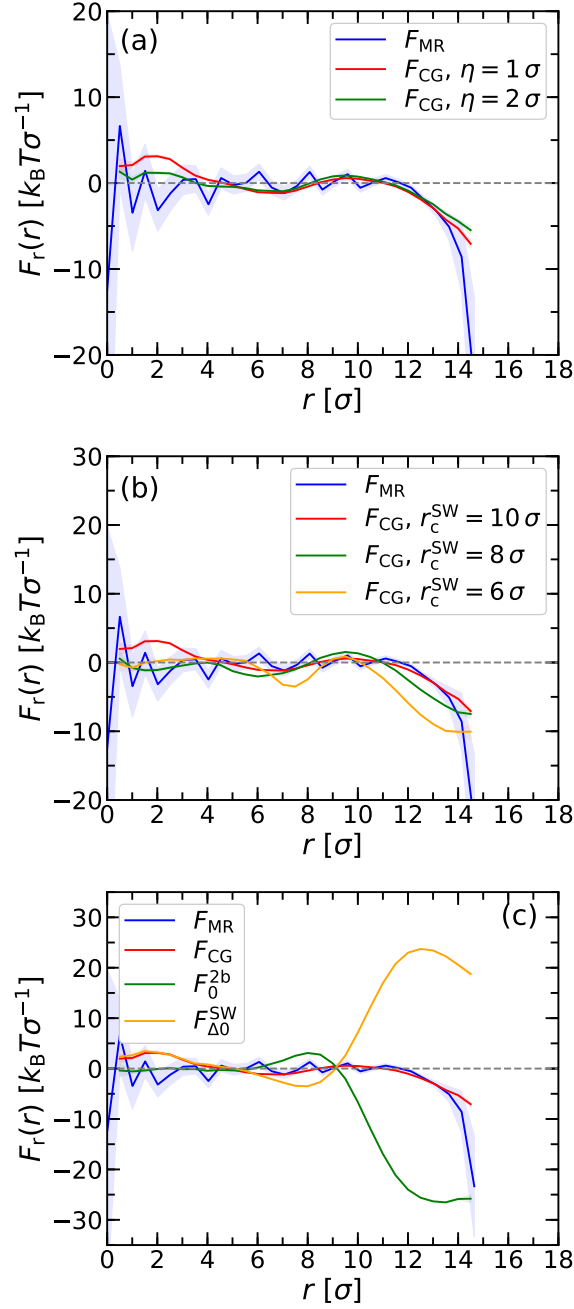


Figure B.7: Radial force profiles along the radial direction r for a droplet with radius $R = 6 R_{g,b}$. Residual force parameterization with pair potential U_0^{2b} . (a) CG Forces at $r_c^{SW} = 10.0 \sigma$ and different parameters η , as indicated. (b) CG Forces at $\eta = \sigma$ and different cutoffs r_c^{SW} , as indicated. (c) Two- and three-body contribution of total CG force at $r_c^{SW} = 10.0 \sigma$ and $\eta = \sigma$. In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

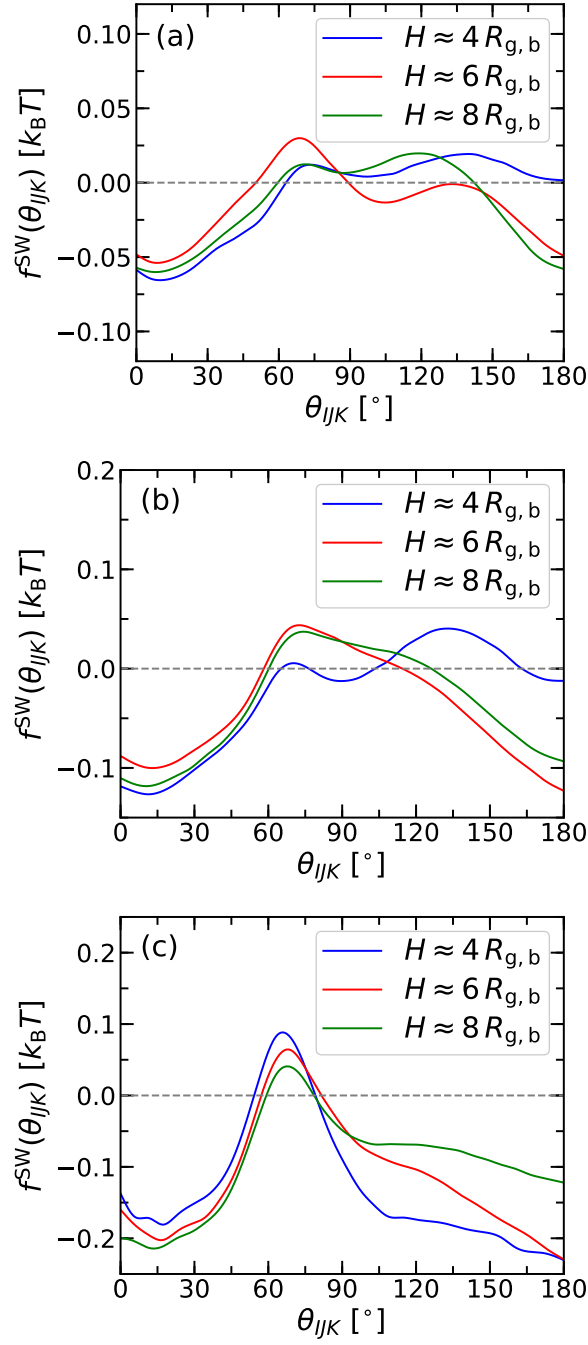


Figure B.8: Angular part $f^{\text{SW}}(\theta_{ijk})$ of $U_{\Delta b}^{\text{SW}}$ at $\eta = \sigma$ for cutoff distance (a) $r_c^{\text{SW}} = 10\sigma$, (b) $r_c^{\text{SW}} = 8\sigma$, and (c) $r_c^{\text{SW}} = 6\sigma$. Data shown for films with different thicknesses H , as indicated.

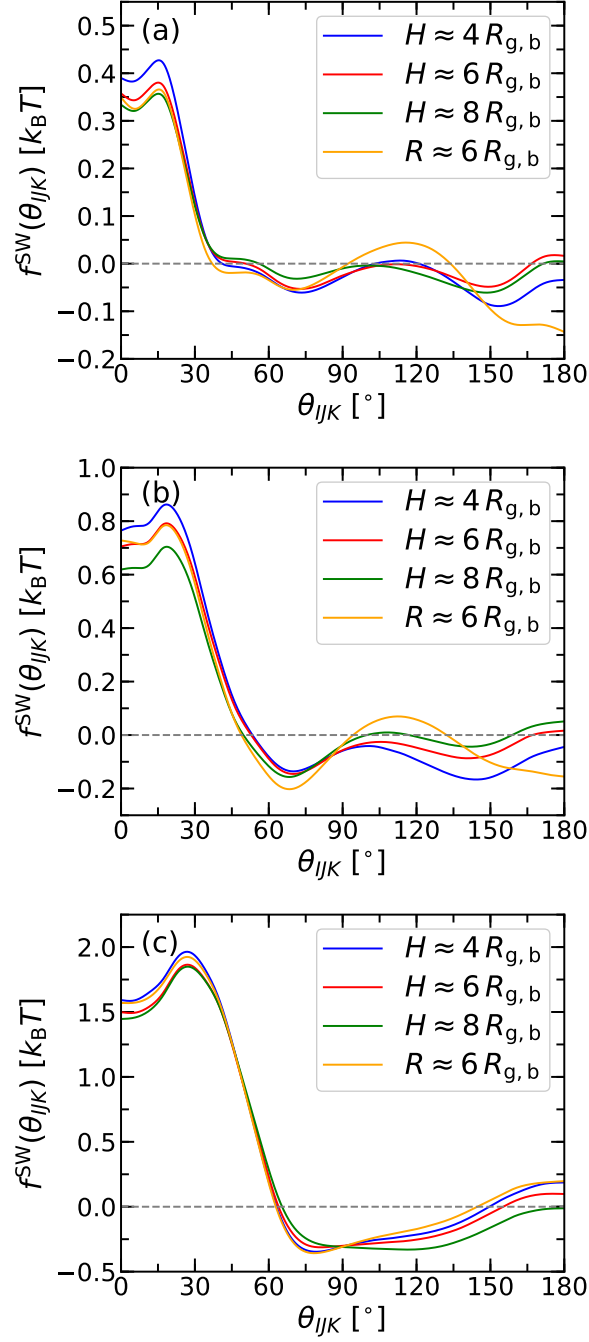


Figure B.9: Angular part $f^{\text{SW}}(\theta_{ijk})$ of $U_{\Delta 0}^{\text{SW}}$ at $\eta = \sigma$ for cutoff distance (a) $r_c^{\text{SW}} = 10\sigma$, (b) $r_c^{\text{SW}} = 8\sigma$, and (c) $r_c^{\text{SW}} = 6\sigma$. Data shown for films with different thicknesses H , as indicated, and droplet with radius $R \approx 6 R_{g,b}$.

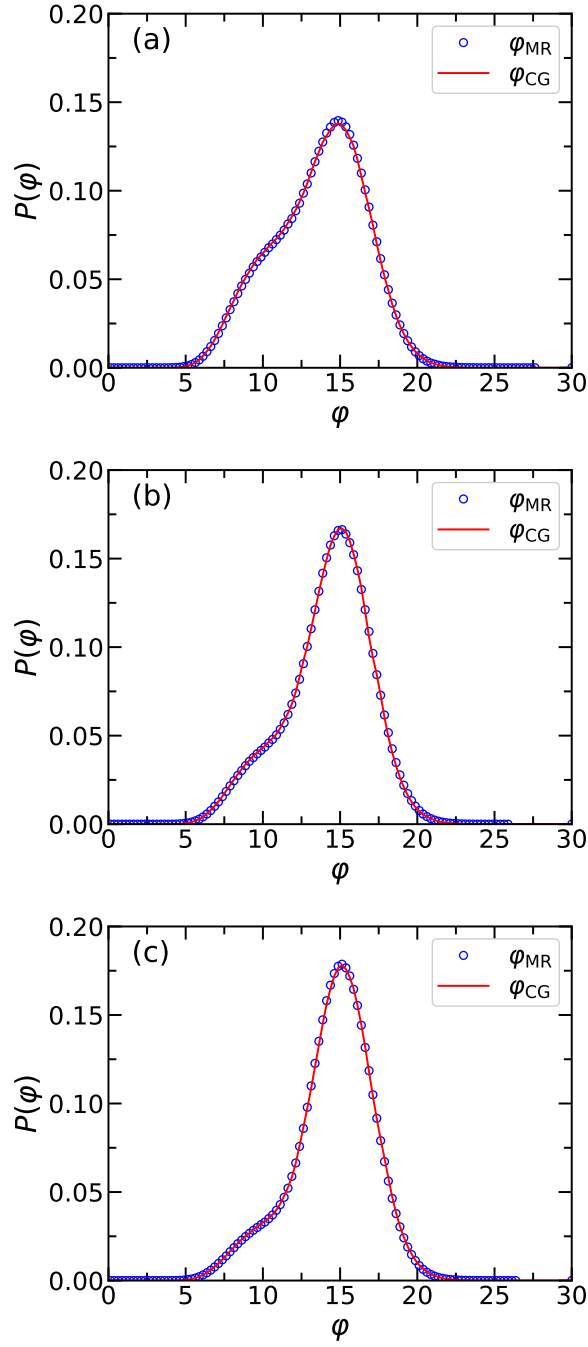


Figure B.10: Local density distribution in reference system and CG system for a film with thickness (a) $H \approx 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$. CG results were taken from simulations using U_0^{2b} and U_0^{LDP} .

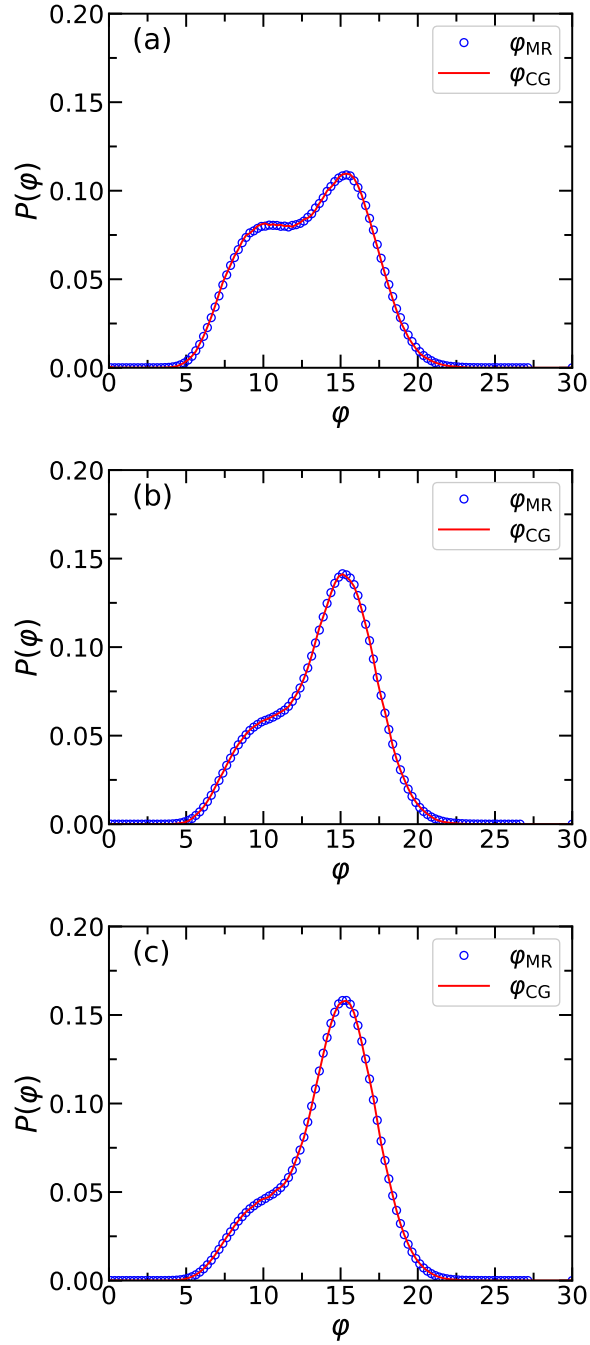


Figure B.11: Local density distribution in reference system and CG system for a droplet with radius (a) $R \approx 4 R_{g,b}$, (b) $R \approx 6 R_{g,b}$, and (c) $R \approx 8 R_{g,b}$. CG results were taken from simulations using U_0^{2b} and U_0^{LDP} .

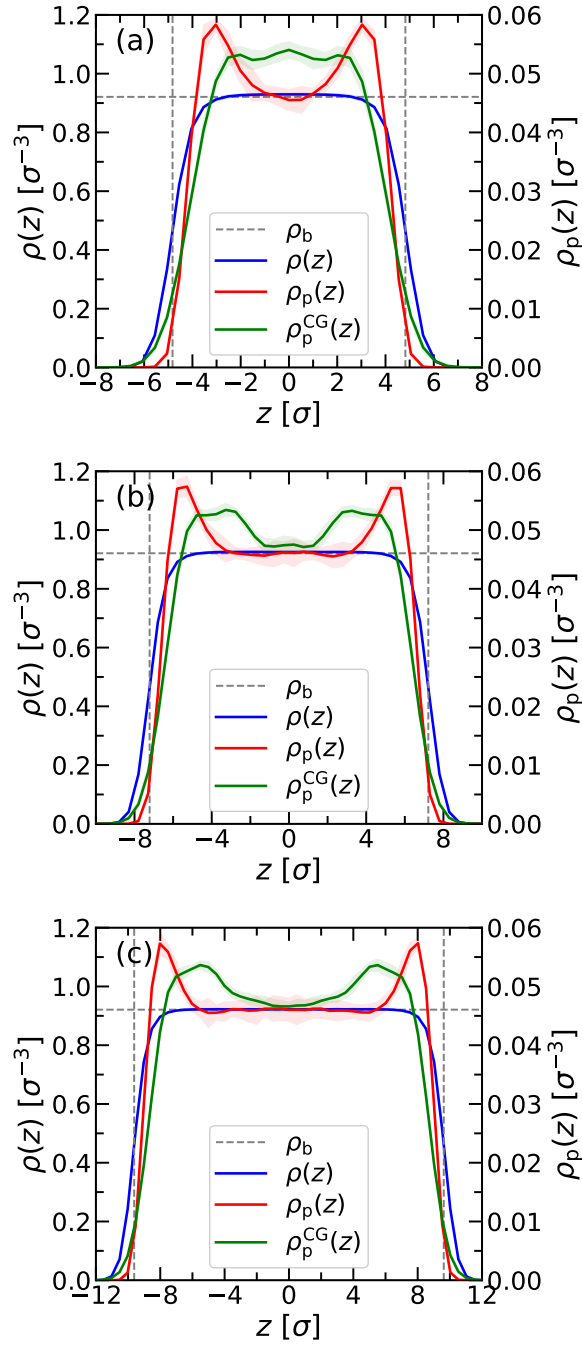


Figure B.12: Density profiles along the z -axis as functions of z for a film with thickness (a) $H \approx 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$. CG results were taken from simulations using U_0^{2b} and U_0^{LDP} . In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

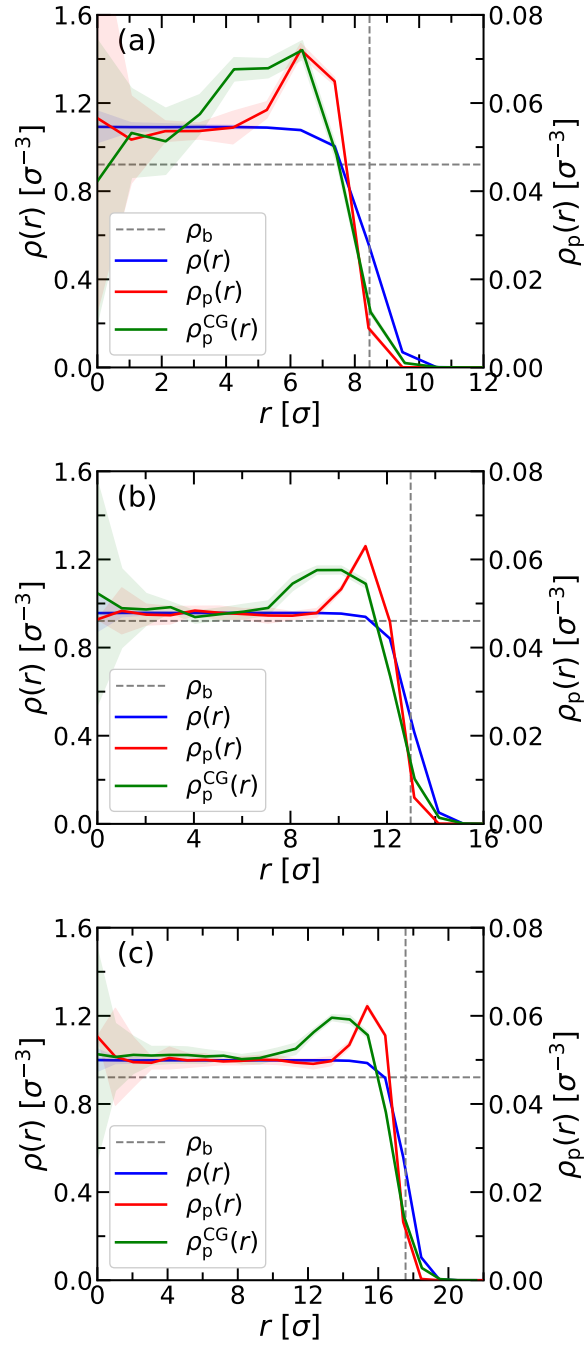


Figure B.13: Radial density profiles as functions of r for a droplet with radius (a) $R \approx 4 R_{g,b}$, (b) $R \approx 6 R_{g,b}$, and (c) $R \approx 8 R_{g,b}$. CG results were taken from simulations using U_0^{2b} and U_0^{LDP} . In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

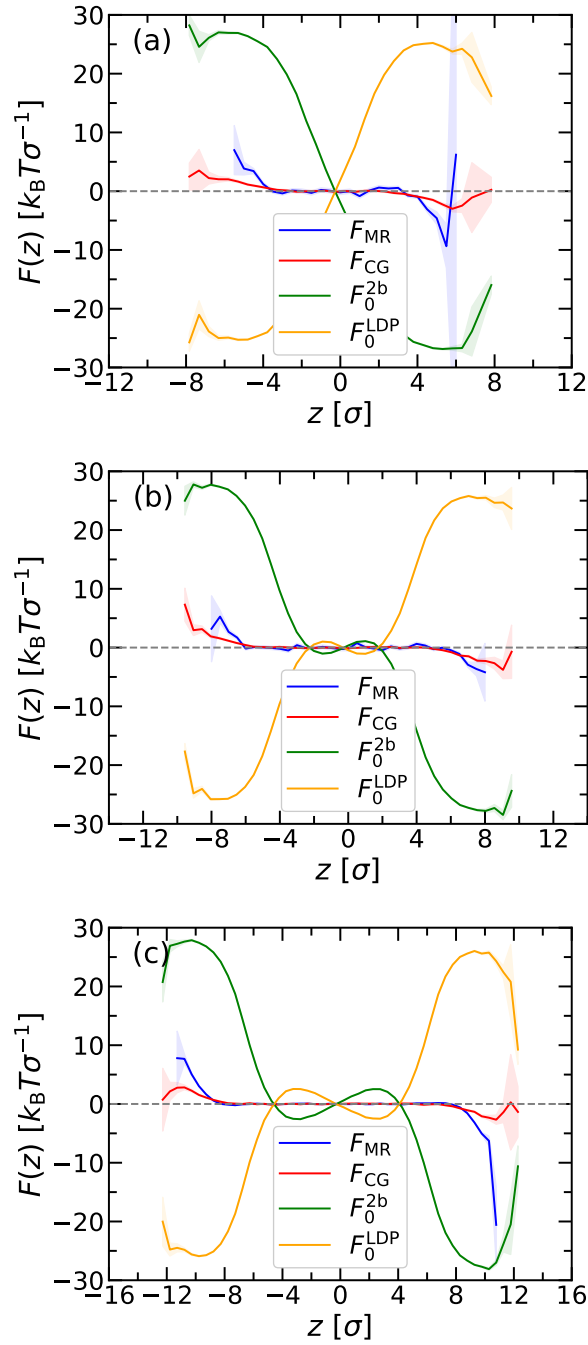


Figure B.14: Force profiles along the z -axis as functions of z for a film with thickness (a) $H \approx 4 R_{g,b}$, (b) $H \approx 6 R_{g,b}$, and (c) $H \approx 8 R_{g,b}$. CG results were taken from simulations using U_0^{2b} and U_0^{LDP} . In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

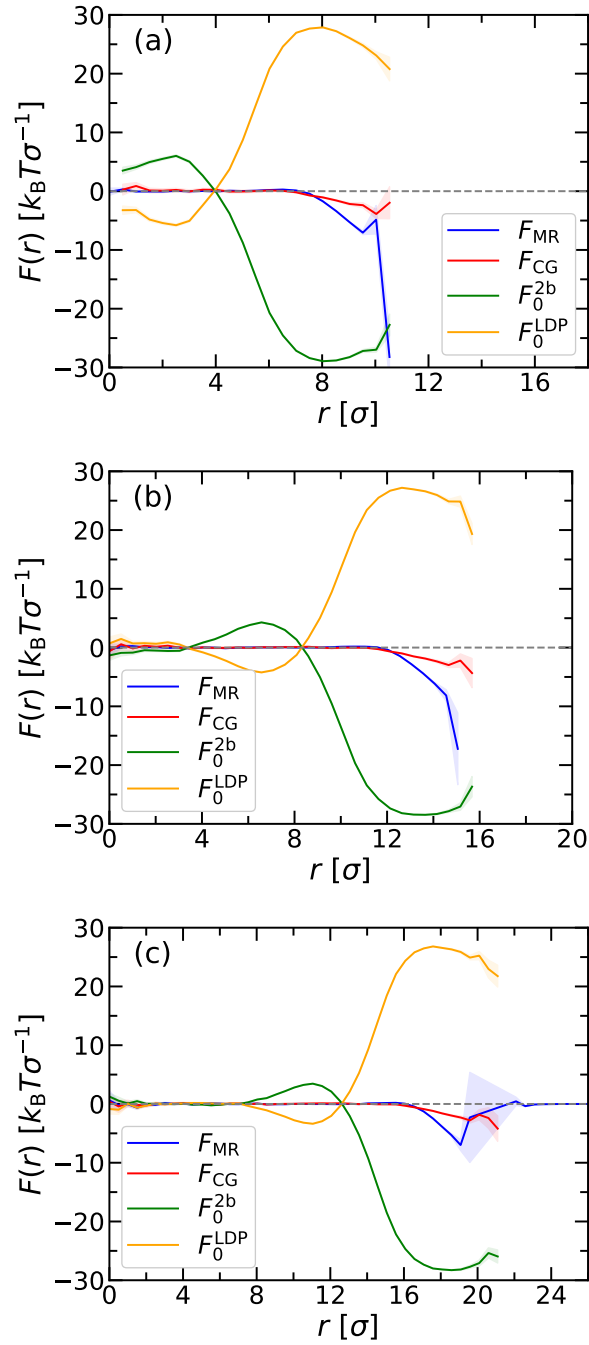


Figure B.15: Radial force profiles as functions of r for a droplet with radius (a) $R \approx 4 R_{g,b}$, (b) $R \approx 6 R_{g,b}$ and (c) $R \approx 8 R_{g,b}$. CG results were taken from simulations using U_0^{2b} and U_0^{LDP} . In all panels, the shaded regions correspond to the measurement uncertainty determined from the standard error of the mean.

CHAPTER C --- Glossary

Glossary for Chap. 2

| Symbol | Meaning | First occurrence |
|-----------------------------------|--|------------------|
| Section 2.1 | | |
| \mathcal{C} | Microscopic configuration | Sec. 2.1 |
| \mathbf{p} | Vector containing all the particles' positions | Sec. 2.1 |
| \mathbf{q} | Vector containing all the particles' momenta | Sec. 2.1 |
| T | Temperature | Sec. 2.1 |
| E | Energy | Sec. 2.1 |
| S | Entropy | Sec. 2.1 |
| P | Pressure | Sec. 2.1 |
| V | Volume | Sec. 2.1 |
| μ | Chemical potential | Sec. 2.1 |
| \mathcal{N} | Number of particles | Sec. 2.1 |
| $\mathcal{H}(\cdot)$ | Hamiltonian function | Sec. 2.1.1 |
| $\mathcal{P}(\cdot)$ | Probability function | Eq. (2.1.1.4) |
| β | Thermodynamic beta | Eq. (2.1.1.4) |
| k_B | Boltzmann constant | Sec. 2.1.1 |
| $\mathcal{Z}(\cdot)$ | Partition function | Eq. (2.1.1.6) |
| $\langle A \rangle$ | Ensemble average of observable A | Eq. (2.1.1.8) |
| $K_{\mathcal{N}}(\cdot)$ | Kinetic energy | Sec. 2.1.1 |
| $V_{\mathcal{N}}(\cdot)$ | Interatomic potential energy | Sec. 2.1.1 |
| $\Phi_{\mathcal{N}}(\cdot)$ | External potential | Sec. 2.1.1 |
| $Z_{\mathcal{N}}(\cdot)$ | Configurational integral | Eq. (2.1.1.12) |
| \mathbf{r} | Set of positions | Sec. 2.1.2 |
| $\rho_{\mathcal{N}}^{(n)}(\cdot)$ | n -particle density | Eq. (2.1.2.1) |
| $\delta(\cdot)$ | Dirac delta distribution | Eq. (2.1.2.1) |
| $g_{\mathcal{N}}^n(\cdot)$ | n -particle distribution function | Eq. (2.1.2.6) |
| ρ | Global density | Sec. 2.1.2 |
| \mathbf{d} | Displacement vector | Sec. 2.1.2 |

Appendix C. Glossary

| | | |
|----------------|--------------------------------|----------------|
| $g(\cdot)$ | Radial distribution function | Eq. (2.1.2.12) |
| $U(\cdot)$ | Pair interaction | Sec. 2.1.2 |
| $S(\cdot)$ | Structure factor | Sec. 2.1.2 |
| \mathbf{k} | Scattering vector | Sec. 2.1.2 |
| B_i | i -th virial coefficient | Sec. 2.1.3 |
| $f(\cdot)$ | Mayer- f -function | Eq. (2.1.3.3) |
| \overline{A} | Time average of observable A | Eq. (2.1.4.1) |
| t | Time | Eq. (2.1.4.1) |

Section 2.2

| | | |
|---------------------|--|----------------|
| M | Molecular weight | Eq. (2.2.0.1) |
| \mathcal{T} | Set of monomer types | Eq. (2.2.0.1) |
| m_α | Mass of monomers of type α | Eq. (2.2.0.1) |
| N_α | Number of monomers of type α | Eq. (2.2.0.1) |
| \mathbf{b} | Bond vector | Sec. 2.2 |
| b | Bond length | Sec. 2.2 |
| P | Persistence length | Sec. 2.2 |
| $\mathbf{R}(\cdot)$ | Polymer contour | Sec. 2.2 |
| L | Polymer length | Sec. 2.2 |
| k | Spring constant | Eq. (2.2.1.1) |
| r_{ij} | Distance between particles i and j | Eq. (2.2.1.1) |
| r_0 | Maximum bond extension | Eq. (2.2.1.1) |
| ε | Interaction strength | Eq. (2.2.1.2) |
| σ | Size parameter | Eq. (2.2.1.2) |
| \mathbf{R}_{ee} | End-to-end distance | Sec. 2.2.2.1 |
| N | Number of monomers in a polymer | Sec. 2.2.2.1 |
| \mathbf{R} | Polymer configuration | Sec. 2.2.2.1 |
| \mathbf{r}_i | Position of monomer i | Sec. 2.2.2.1 |
| \mathcal{R} | Set of all polymer configurations | Eq. (2.2.2.3) |
| \cdot | Standard scalar product | Sec. 2.2.2.1 |
| δ_{ij} | Kronecker delta of i and j | Eq. (2.2.2.8) |
| Θ_{ij} | Angle between bond vectors \mathbf{b}_i and \mathbf{b}_j | Eq. (2.2.2.14) |
| b_k | Kuhn length | Sec. 2.2.2.1 |
| C'_i | Flory coefficient | Eq. (2.2.2.47) |
| C_N | Flory's characteristic ratio for chain of length N | Eq. (2.2.2.47) |
| N_k | Number of Kuhn segments | Sec. 2.2.2.2 |
| R_{\max} | Maximum end-to-end distance | Sec. 2.2.2.2 |

| | | |
|---------------------------|---|----------------|
| R_g^2 | Squared radius of gyration | Sec. 2.2.2.3 |
| \mathbf{R}_{com} | Center of mass position | Eq. (2.2.2.80) |
| \mathbf{G} | Gyration tensor | Eq. (2.2.2.84) |
| \otimes | Dyadic product | Eq. (2.2.2.84) |
| $\Delta \mathbf{d}_i$ | Displacement of monomer i from polymer COM | Eq. (2.2.2.84) |
| $U(\cdot)$ | Pair potential between monomers | Sec. 2.2.3.1 |
| \mathbf{r} | Distance vector between monomers | Sec. 2.2.3.1 |
| $f(\cdot)$ | Mayer- f -function | Eq. (2.2.3.1) |
| v | Excluded volume | Eq. (2.2.3.2) |
| d_{BH} | Barker-Henderson diameter | Eq. (2.2.3.6) |
| σ_0 | Distance at which the pair potential vanishes | Eq. (2.2.3.6) |
| R | Square root of the mean squared end-to-end distance | Eq. (2.2.3.7) |
| ν | Scaling exponent of size of chains with $(N - 1)$ | Eq. (2.2.3.10) |
| n_T | Number of monomers contained in a thermal blob | Eq. (2.2.3.17) |
| ξ_T | Size of a thermal blob | Eq. (2.2.3.18) |

Section 2.3

| | | |
|---|--|----------------|
| $\mathcal{P}_{\text{B}}(\cdot)$ | Canonical probability distribution of configurations | Sec. 2.3.1 |
| $\mathcal{P}_{\text{S}}(\cdot)$ | Sampled probability distribution of configurations | Sec. 2.3.1 |
| $\mathcal{C} \sim \mathcal{P}_{\text{S}}$ | Configurations \mathcal{C} are sampled according to \mathcal{P}_{S} | Eq. (2.3.1.4) |
| \mathbf{P} | Phase space | Eq. (2.3.1.5) |
| S_i | State i | Sec. 2.3.1 |
| ω | Transition matrix | Eq. (2.3.1.8) |
| $\text{Acc}(\cdot, \cdot)$ | Metropolis acceptance ratio | Eq. (2.3.1.10) |
| \mathbf{F} | Force | Sec. 2.3.2 |
| \mathbf{a} | Acceleration | Sec. 2.3.2 |
| Δt | Timestep | Sec. 2.3.2.1 |
| $\mathbf{J}_{2n \times 2n}$ | $(2n \times 2n)$ -dimensional symplectic unit matrix | Eq. (2.3.2.2) |
| $\nabla_{2\mathcal{N}}$ | $2\mathcal{N}$ -dimensional gradient | Eq. (2.3.2.9) |
| $\boldsymbol{\eta}_i(t)$ | Uniform random force acting on particle i | Sec. 2.3.2.2 |
| $\mathbf{F}_i^{\text{frict}}$ | Friction force acting on particle i | Sec. 2.3.2.2 |
| γ | Friction coefficient | Sec. 2.3.2.2 |
| $\mathcal{M}(\cdot)$ | Memory kernel | Sec. 2.3.2.2 |
| L_α | Size of simulation box in direction $\alpha \in \{x, y, z\}$ | Sec. 2.3.3 |
| \mathbf{I}_i | Periodic image indices of particle i | Sec. 2.3.3 |
| \mathbf{r}_i | Absolute position of particle i | Fig. 2.4 |
| $\bar{\mathbf{r}}_i$ | Relative position inside the box of particle i | Fig. 2.4 |

Appendix C. Glossary

| | | |
|-----------------------|--|---------------|
| \mathbf{d}_{ij} | Minimal distance between particles i and j | Fig. 2.4 |
| $U^{nb}(\cdot)$ | n -body potential energy function | Sec. 2.3.4 |
| $G^{nb}(\cdot)$ | n -body embedding function | Sec. 2.3.4 |
| $\omega^{nb}(\cdot)$ | n -body weight function | Sec. 2.3.4 |
| $U^{nb-LDP}(\cdot)$ | n -body LDP function | Eq. (2.3.4.4) |
| $\varphi^{nb}(\cdot)$ | n -body local density | Eq. (2.3.4.4) |
| σ | Unit of length | Sec. 2.3.5 |
| ε | Unit of energy/interaction strength | Sec. 2.3.5 |
| m | Unit of mass | Sec. 2.3.5 |

Section 2.4

| | | |
|----------------------------------|---|---------------|
| $g^j(r)$ | Radial distribution function in j -th iteration | Sec. 2.4.1 |
| $g_{\text{ref}}(r)$ | Reference radial distribution function | Sec. 2.4.1 |
| $U^j(\cdot)$ | Potential energy function in j -th iteration | Eq. (2.4.1.1) |
| α | Scaling factor in update rule | Eq. (2.4.1.1) |
| n_{CG} | Number of coarse-grained beads | Sec. 2.4.2 |
| \mathbf{R} | CG configuration | Sec. 2.4.2 |
| $\mathcal{V}^{\text{CG}}(\cdot)$ | Coarse-grained interaction function | Sec. 2.4.2 |
| \mathbf{r} | Atomistic reference configuration | Sec. 2.4.2 |
| n | Number of atomistic particles | Sec. 2.4.2 |
| $\mathbf{M}(\cdot)$ | Mapping function from atomistic to CG configurations | Sec. 2.4.2 |
| \mathbf{F}^{CG} | Coarse-grained forces | Sec. 2.4.2 |
| χ^2 | MS-CG residual | Eq. (2.4.2.3) |
| $\mathcal{Q}(\cdot)$ | Prob. distribution of CG config. | Sec. 2.4.3 |
| $\mathcal{Q}'(\cdot)$ | Prob. distribution of atom. config. with CG Hamiltonian | Sec. 2.4.3 |
| $\Omega(\cdot)$ | Degeneracy of CG config. | Eq. (2.4.3.1) |
| $\mathcal{P}(\cdot)$ | Probability distribution of atomistic config. | Sec. 2.4.3 |
| $S_{\text{rel}}(\cdot)$ | Relative entropy | Eq. (2.4.3.3) |
| A | Helmholtz free energy | Eq. (2.4.3.6) |
| λ_i^j | i -th parameter of the potential in the j -th iteration | Sec. 2.4.3 |

Section 2.5

| | | |
|--------------|----------------------------|--------------|
| E | Experience | Sec. 2.5.1 |
| T | Task | Sec. 2.5.1 |
| P | Performance measure | Sec. 2.5.1 |
| Ω | Mapping function | Sec. 2.5.1.1 |
| \mathbf{X} | Input (/target) properties | Sec. 2.5.1.1 |

| | | |
|---------------------------------------|---|----------------|
| \mathbf{Y} | Target properties | Sec. 2.5.1.1 |
| N | Number of examples | Sec. 2.5.1.1 |
| Θ | Parametrization of ML model | Sec. 2.5.1.1 |
| M | Performance metric | Sec. 2.5.1.1 |
| \mathbf{S} | State | Sec. 2.5.1.1 |
| \mathbf{a} | Action | Sec. 2.5.1.1 |
| $\langle \mathbf{G} \rangle_{\Theta}$ | Expected discounted return of policy Θ | Sec. 2.5.1.1 |
| $R(\cdot)$ | Reward function | Sec. 2.5.1.1 |
| r | Reward | Sec. 2.5.1.1 |
| \mathcal{S}_{Θ} | Set of states observed under policy Θ | Eq. (2.5.1.4) |
| γ | Discount factor | Eq. (2.5.1.4) |
| $z(\cdot)$ | Activation of a neuron | Sec. 2.5.2.1 |
| ω | Weight vector/matrix | Eq. (2.5.2.1) |
| $a(\cdot)$ | Nonlinear activation function | Eq. (2.5.2.1) |
| b | Bias | Eq. (2.5.2.1) |
| $\mathbf{z}(\cdot)$ | Activation of a layer of neurons | Sec. 2.5.2.1 |
| \mathbf{b} | Bias vector | Sec. 2.5.2.1 |
| $\mathbf{O}(\cdot)$ | Output of a neural network | Sec. 2.5.2.1 |
| k | Kernel size | Eq. (2.5.2.17) |
| \mathcal{K} | Convolutional kernel with learnable weights | Eq. (2.5.2.17) |
| \star | Cross-correlation | Eq. (2.5.2.18) |
| $*$ | Convolution | Eq. (2.5.2.21) |
| $\mathcal{L}(\cdot)$ | Loss function | Sec. 2.5.2.3 |
| α | Learning rate | Eq. (2.5.2.22) |
| $L_1(\cdot)$ | L_1 -regularization function | Eq. (2.5.3.1) |
| $L_2(\cdot)$ | L_2 -regularization function | Eq. (2.5.3.2) |

Glossary for Chap. 3

| Symbol | Meaning | First occurrence |
|-------------|----------------|------------------|
| Section 3.1 | | |
| $U(\cdot)$ | Pair potential | Sec. 3.1 |

Appendix C. Glossary

| | | |
|-------------------------|---|---------------|
| $g(\cdot)$ | Radial pair distribution function | Sec. 3.1 |
| Section 3.2 | | |
| β | Thermodynamic beta | Sec. 3.2 |
| k_B | Boltzmann constant | Sec. 3.2 |
| T | Temperature | Sec. 3.2 |
| y | Target | Sec. 3.2 |
| \hat{y} | Prediction | Sec. 3.2 |
| $l(\cdot)$ | Basic loss function | Eq. (3.2.1.1) |
| $L(\cdot)$ | Loss of entire output vector | Eq. (3.2.1.4) |
| d | Dimensionality of target and prediction vector | Eq. (3.2.1.4) |
| $\langle \cdot \rangle$ | Average over multiple output vectors | Sec. 3.2 |
| α | Weight factor | Eq. (3.2.1.5) |
| L_Δ | Discretized Laplace loss | Eq. (3.2.1.5) |
| L_k | k -distance correlation loss | Eq. (3.2.1.5) |
| P | (Target) Pressure | Sec. 3.2.2 |
| σ | Reduced unit of length | Sec. 3.2.2 |
| ε | Reduced unit of energy | Sec. 3.2.2 |
| m | Reduced unit of mass | Sec. 3.2.2 |
| τ | Reduced unit of time | Sec. 3.2.2 |
| r_{cut} | Cutoff radius | Sec. 3.2.2 |
| n | Number of base points of spline functions | Sec. 3.2.2 |
| $F(\cdot)$ | Pair force function | Sec. 3.2.2 |
| Δt | Timestep | Sec. 3.2.2 |
| ρ | Particle number density | Sec. 3.2.2 |
| N | Number of particles | Sec. 3.2.2 |
| B_2^* | Effective second virial coefficient | Sec. 3.3.1 |
| \mathcal{B}_n | Set of virial coefficients up to order n | Sec. 3.3.1 |
| B_i | i -th virial coefficient | Sec. 3.3.1 |
| $\bar{U}(\cdot)$ | Clipped pair potential | Sec. 3.3.1 |
| $U_{\text{cut}}(\cdot)$ | Cutoff value for pair potential | Sec. 3.3.1 |
| P_0 | Pressure estimate using second virial coefficient | Sec. 3.3.1 |
| f | Mayer- f -function | Sec. 3.3.1 |
| \hat{P} | Predicted pressure | Sec. 3.3.1 |
| R^2 | Coefficient of determination | Sec. 3.3.1 |
| $U_0(\cdot)$ | Low density approximation of pair potential | Eq. (3.3.2.1) |
| $\hat{U}(\cdot)$ | Predicted pair potential | Eq. (3.3.2.1) |

| | | |
|------------------|--|------------|
| $\hat{g}(\cdot)$ | Radial distribution function resulting from prediction | Sec. 3.3.2 |
| $p(\cdot)$ | Probability density function of losses | Sec. 3.3.2 |
| δ | Width of distribution | Sec. 3.3.2 |
| $g_0(\cdot)$ | Radial distribution function resulting from $U_0(\cdot)$ | Sec. 3.3.2 |

Section 3.3.3

| | | |
|--------------------|--|------------|
| $U_{LJ}(\cdot)$ | Lennard-Jones potential | Sec. 3.3.3 |
| ε_{LJ} | Lennard-Jones interaction strength | Sec. 3.3.3 |
| σ_{LJ} | Lennard-Jones size parameter | Sec. 3.3.3 |
| N_p | Number of polymers | Sec. 3.3.3 |
| N | Degree of polymerization | Sec. 3.3.3 |
| k | FENE bond strength | Sec. 3.3.3 |
| r_0 | Maximum bond extension | Sec. 3.3.3 |
| $\hat{g}^i(r)$ | Radial distribution function in i -th iteration of IBI | Sec. 3.3.3 |

Glossary for Chap. 4

| Symbol | Meaning | First occurrence |
|--------------------------|---------------------------------------|------------------|
| Section 4.2 | | |
| N_p | Number of polymers | Sec. 4.2.1 |
| N | Degree of polymerization | Sec. 4.2.1 |
| $U^{LJ}(\cdot)$ | Lennard-Jones potential | Eq. (4.2.1.1) |
| r_{ij} | Distance between monomers i and j | Eq. (4.2.1.1) |
| ε | Lennard-Jones interaction strength | Eq. (4.2.1.1) |
| k_B | Boltzmann constant | Sec. 4.2.1 |
| T | Temperature | Sec. 4.2.1 |
| σ | Lennard-Jones size parameter | Eq. (4.2.1.1) |
| r_c | Cutoff radius | Eq. (4.2.1.1) |
| $U^{\text{FENE}}(\cdot)$ | FENE potential | Eq. (4.2.1.2) |
| k | FENE bond strength | Eq. (4.2.1.2) |
| r_0 | Maximum bond extension | Eq. (4.2.1.1) |
| \mathcal{N} | Number of particles | Sec. 4.2.1 |
| ξ | Langevin friction coefficient | Sec. 4.2.1 |
| m | Monomer mass | Sec. 4.2.1 |
| τ | Unit of time | Sec. 4.2.1 |
| Δt | Timestep | Sec. 4.2.1 |

Appendix C. Glossary

| | | |
|-----------------------------------|---|---------------|
| L | Size of simulation box | Sec. 4.2.1 |
| P | Pressure | Sec. 4.2.1 |
| ρ_b | Average monomer number density in bulk | Sec. 4.2.1 |
| \mathbf{G} | Average radius of gyration tensor | Eq. (4.2.1.3) |
| $\Delta \mathbf{r}_i$ | Vector from polymer center of mass to monomer i | Eq. (4.2.1.3) |
| \otimes | Dyadic product | Eq. (4.2.1.3) |
| R_g | Radius of gyration | Sec. 4.2.1 |
| H | Thickness of film | Sec. 4.2.1 |
| R | Radius of droplet | Sec. 4.2.1 |
| $\rho(\cdot)$ | Monomer number density profile | Sec. 4.2.1 |
| T_c | Critical temperature | Sec. 4.2.1 |
| $\rho_p(\cdot)$ | Number density profile of polymer centers of mass | Sec. 4.2.1 |
| γ | Surface tension | Sec. 4.2.1 |
| U_b^{2b} | Two-body CG interactions from bulk | Sec. 4.2.2.1 |
| U_0^{2b} | Two-body CG interactions from infinite dilution | Sec. 4.2.2.1 |
| r_c^{2b} | Cutoff radius of two-body CG interactions | Sec. 4.2.2.1 |
| $f^{2b}(\cdot)$ | Two-body CG force | Sec. 4.2.2.1 |
| Δr^{2b} | Grid spacing of CG forces | Sec. 4.2.2.1 |
| K | Number of coefficients of CG forces | Sec. 4.2.2.1 |
| λ_i | i -th spline coefficient | Sec. 4.2.2.1 |
| N_s | Number of snapshots | Sec. 4.2.2.1 |
| $\Delta r_{\text{tab}}^{2b}$ | Grid spacing of tabulated interactions | Sec. 4.2.2.1 |
| $f^{\text{sm}}(\cdot)$ | Smoothing function | Sec. 4.2.2.1 |
| r^{sm} | Smoothing distance | Sec. 4.2.2.1 |
| $U^{\text{SW}}(\cdot)$ | Three-body Stillinger-Weber potential | Eq. (4.2.2.2) |
| r_{IJ} | Distance between CG beads I and J | Eq. (4.2.2.2) |
| $f^{\text{SW}}(\cdot)$ | Angular interaction term | Eq. (4.2.2.2) |
| θ_{IJK} | Angle between CG beads I , J and K | Eq. (4.2.2.2) |
| η | Control parameter for steepness of potential | Eq. (4.2.2.2) |
| r_c^{SW} | Stillinger-Weber cutoff radius | Eq. (4.2.2.2) |
| $\Delta \mathbf{f}$ | Residual forces of the MR simulations | Eq. (4.2.2.2) |
| $\mathbf{f}^{2b}(\cdot)$ | Pairwise CG force | Sec. 4.2.2.2 |
| $\mathbf{f}_{\text{MR}}(\cdot)$ | Pairwise MR force | Sec. 4.2.2.2 |
| $U_{\Delta b}^{\text{SW}}(\cdot)$ | Three-body CG interaction from bulk | Sec. 4.2.2.2 |
| $U_{\Delta 0}^{\text{SW}}(\cdot)$ | Three-body CG interaction from infinite dilution | Sec. 4.2.2.2 |
| $\omega(\cdot)$ | Radial weight function of LDP | Sec. 4.2.2.3 |
| $\rho_{\text{cloud}}(\cdot)$ | Monomer cloud density around polymer COM | Sec. 4.2.2.3 |

| | | |
|--|--|---------------|
| $\delta(\cdot)$ | Dirac delta distribution | Eq. (4.2.2.4) |
| \mathbf{r}_{com} | Polymer COM position | Eq. (4.2.2.4) |
| U_I^{LDP} | Potential energy of CG particle I due to LDP | Eq. (4.2.2.6) |
| $G(\cdot)$ | Embedding function | Eq. (4.2.2.6) |
| φ | Local density of (fictitious) monomer pairs | Eq. (4.2.2.6) |
| λ_i | i -th node of cardinal B-spline function | Sec. 4.2.2.3 |
| $\Delta\varphi$ | Grid spacing of cardinal B-spline function | Sec. 4.2.2.3 |
| S_{rel} | Relative entropy | Eq. (4.2.2.8) |
| $\langle \cdot \rangle_{\text{CG}}(\cdot)$ | Ensemble average in CG system | Eq. (4.2.2.8) |
| $\langle \cdot \rangle_{\text{MR}}(\cdot)$ | Ensemble average in MR system | Eq. (4.2.2.8) |
| $U_{\text{CG}}(\cdot)$ | CG potential function | Eq. (4.2.2.8) |
| $U_{\text{MR}}(\cdot)$ | CG potential function | Eq. (4.2.2.8) |
| $A_{\text{CG}}(\cdot)$ | CG Helmholtz free energy | Eq. (4.2.2.8) |
| $A_{\text{MR}}(\cdot)$ | CG Helmholtz free energy | Eq. (4.2.2.8) |
| S_{map} | Relative entropy contribution from mapping | Eq. (4.2.2.8) |
| α | Scaling factor in update rule | Sec. 4.2.2.3 |
| $\Delta\lambda_i^j$ | Change of i -th parameter in j -th iteration | Sec. 4.2.2.3 |

Section 4.3

| | | |
|-----------------------------|---|------------|
| σ_p | Characteristic length scale of interaction | Sec. 4.3.3 |
| $P(\cdot)$ | Probability density function of local density | Sec. 4.3.3 |
| $U^{\text{LDP},2b}(\cdot)$ | Additive pair potential from linear regime of LDP | Sec. 4.3.3 |
| $\rho_p^{\text{CG}}(\cdot)$ | Number density profile of CG beads | Fig. 4.9 |

Glossary for Chap. 5

| Symbol | Meaning | First occurrence |
|------------------------|--|------------------|
| Section 5.1 | | |
| N | Degree of polymerization | Sec. 5.1.1 |
| $S(\cdot)$ | Smoothing function | Sec. 5.1.1 |
| $U^{\text{LJ}}(\cdot)$ | Lennard-Jones potential | Eq. (5.1.1.1) |
| r_{ij} | Distance between particles i and j | Eq. (5.1.1.1) |

Appendix C. Glossary

| | | |
|-----------------------------|--|---------------|
| t_i | Type of particle i | Eq. (5.1.1.1) |
| $\varepsilon_{\alpha\beta}$ | LJ interaction strength between types α and β | Eq. (5.1.1.1) |
| $\sigma_{\alpha\beta}$ | LJ size parameter between types α and β | Eq. (5.1.1.1) |
| r_c | Cutoff radius | Eq. (5.1.1.1) |
| r_{on} | Onset radius of smoothing | Eq. (5.1.1.2) |
| k_B | Boltzmann constant | Tab. 5.1 |
| m | Mass/mass scale | Tab. 5.1 |
| u | Atomic mass unit | Tab. 5.1 |
| $U^{\text{bond}}(\cdot)$ | Harmonic bond potential | Eq. (5.1.1.5) |
| k | Spring constant | Eq. (5.1.1.5) |
| r_0 | Bond rest length | Eq. (5.1.1.5) |
| T | Temperature | Sec. 5.1.1 |
| $U^{\text{ang}}(\cdot)$ | Harmonic angle potential | Eq. (5.1.1.6) |
| θ_{ijk} | Angle between neighboring particles i, j and k | Eq. (5.1.1.6) |
| k_a | Angle potential strength | Eq. (5.1.1.6) |
| θ_0 | Rest angle | Eq. (5.1.1.6) |
| $U^{\text{dih}}(\cdot)$ | Dihedral OPLS potential | Eq. (5.1.1.7) |
| ϕ_{ijkl} | Dihedral angle of neighboring particles i, j, k and l | Eq. (5.1.1.7) |
| C_1, C_2, C_3 | OPLS coefficients | Eq. (5.1.1.7) |
| \mathcal{N} | Number of particles | Sec. 5.1.1 |
| N_p | Number of polymers | Sec. 5.1.1 |
| ξ | Langevin friction coefficient | Sec. 5.1.1 |
| σ | Length scale | Sec. 5.1.1 |
| ε | Energy scale | Sec. 5.1.1 |
| τ | Time scale | Sec. 5.1.1 |
| Δt | Timestep | Sec. 5.1.1 |
| P | Pressure | Sec. 5.1.1 |
| $\rho(\cdot)$ | Monomer number density profile | Sec. 5.1.1 |
| $\rho_p(\cdot)$ | Number density profile of polymer centers of mass | Sec. 5.1.1 |
| $\rho_b(\cdot)$ | Monomer number density in bulk | Sec. 5.1.1 |
| \mathbf{r}_{com} | Center of mass position | Eq. (5.1.2.1) |
| M | Total polymer mass | Eq. (5.1.2.1) |
| m_i | Mass of monomer i | Eq. (5.1.2.1) |
| \mathbf{r}_i | Position of monomer i | Eq. (5.1.2.1) |
| $G(\cdot)$ | Embedding function | Sec. 5.1.2 |

Section 5.2

| | | |
|------------------------------|--|---------------|
| φ | Local density of (fictitious) monomer pairs | Sec. 5.2 |
| $g_b(\cdot)$ | Radial distribution function in bulk | Sec. 5.2 |
| $\langle \cdot \rangle_b$ | Ensemble average in bulk | Sec. 5.2 |
| $\omega(\cdot)$ | Radial weight function of LDP | Eq. (5.2.0.2) |
| $P(\cdot)$ | Probability of segment displacement from COM | Sec. 5.2 |
| \mathbf{r} | Segment displacement from COM | Sec. 5.2 |
| b | Bond length | Sec. 5.2 |
| $\rho_{\text{cloud}}(\cdot)$ | Monomer cloud density around polymer COM | Fig. 5.2 |

Section 5.3

| | | |
|---------------------------------|---|----------|
| $\langle \Delta\varphi \rangle$ | Spacing of observed peaks of local density distribution | Sec. 5.3 |
| R | Ratio of peak spacing and maximum of weight function | Sec. 5.3 |

CHAPTER **D** --- Revisions and Improvements

After receiving feedback and evaluation on the initial submission of this dissertation, the following revisions have been made to further enhance the theoretical discussions in this work:

- The general description of polymers in Sec. 2.2 has been refined to clarify the distinction between chemical monomers and monomers of polymer models.
- The discussion of Flory's characteristic ratio in Sec. 2.2.2.1 has been expanded to cover the general case for ideal models, and an example for the freely jointed chain has been included.
- The description of the theta temperature at the end of Sec. 2.2.2.1 has been omitted.
- The treatment of the Kuhn segmentation in Sec. 2.2.2.2 has been generalized to apply to all ideal polymer models, and an example has been provided for the freely jointed chain.
- Section 2.2.3 has been newly added, encompassing a detailed description of the scaling behavior of polymer chains, the concept of excluded volume, and an explanation of the theta temperature.
- The requirements for trial moves in Monte Carlo Simulations in Sec. 2.3.1 have been stated with greater precision.
- The wording of the statement about Hamilton's equations in Sec. 2.3.2.1 below Eq. (2.3.2.9) has been rephrased for improved clarity.
- The table of contents, the glossary and the bibliography have been expanded in accordance to the added section mentioned above.

Additionally, minor editorial changes have been applied to enhance the overall readability and cohesiveness of the dissertation.

Bibliography

- [1] R. Brown, “XXVII. A brief account of microscopical observations made in the months of June, July and August 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies”, *The Philosophical Magazine* **4**, 161–173 (1828).
- [2] T. L. Carus, *De rerum natura*, 1st c. BC.
- [3] H. Braconnot, “De la transformation de plusieurs substances végétales en un principe nouveau”, *Ann. Chim. Phys.* **52**, 290–294 (1833).
- [4] C. F. Schönbein, “Notiz über eine Veränderung der Pflanzenfaser und einiger andern organischen Substanzen”, *Bericht über die Verhandlungen der Naturforschenden Gesellschaft in Basel* **7**, 26–27 (1846).
- [5] V. Gold, *The IUPAC Compendium of Chemical Terminology* (International Union of Pure and Applied Chemistry (IUPAC), 2019).
- [6] H. K. Onnes, “Expression of the equation of state of gases and liquids by means of series”, in *KNAW Proceedings* (1902), 125–147.
- [7] G. S. Grest and K. Kremer, “Molecular dynamics simulation for polymers in the presence of a heat bath”, *Phys. Rev. A* **33**, 3628–3631 (1986).
- [8] K. Kremer and G. S. Grest, “Dynamics of entangled linear polymer melts: A molecular-dynamics simulation”, *J. Chem. Phys.* **92**, 5057–5086 (1990).
- [9] M. Bishop, M. H. Kalos, and H. L. Frisch, “Molecular dynamics of polymeric systems”, *J. Chem. Phys.* **70**, 1299–1304 (1979).
- [10] J. D. Weeks, D. Chandler, and H. C. Andersen, “Role of Repulsive Forces in Determining the Equilibrium Structure of Simple Liquids”, *J. Chem. Phys.* **54**, 5237–5247 (1971).
- [11] J. E. Jones and S. Chapman, “On the determination of molecular fields.—I. From the variation of the viscosity of a gas with temperature”, *Proc. R. Soc. London A.* **106**, 441–462 (1924).
- [12] J. E. Jones and S. Chapman, “On the determination of molecular fields.—II. From the equation of state of a gas”, *Proc. R. Soc. London A.* **106**, 463–477 (1924).
- [13] J. E. Lennard-Jones, “Cohesion”, *Proc. Phys. Soc.* **43**, 461–482 (1931).

Bibliography

- [14] W. Kuhn, "Beziehungen zwischen Molekülgröße, statistischer Molekülgestalt und elastischen Eigenschaften hochpolymerer Stoffe", *Kolloid. Z.* **76**, 258–271 (1936).
- [15] J. Rudnick and G. Gaspari, "The Shapes of Random Walks", *Science* **237**, 384–389 (1987).
- [16] J. A. Barker and D. Henderson, "Perturbation Theory and Equation of State for Fluids. II. A Successful Theory of Liquids", *J. Chem. Phys.* **47**, 4714–4721 (1967).
- [17] M. Rubinstein and R. H. Colby, *Polymer physics* (Oxford University Press, 2003).
- [18] G.-L. L. Comte de Buffon, *Histoire naturelle, générale et particulière Supplément 4*, 1777.
- [19] N. Metropolis and S. Ulam, "The Monte Carlo Method", *J. Amer. Statist. Assoc.* **44**, 335–341 (1949).
- [20] V. I. Manousiouthakis and M. W. Deem, "Strict detailed balance is unnecessary in Monte Carlo simulation", *J. Chem. Phys.* **110**, 2753–2756 (1999).
- [21] B. J. Alder and T. E. Wainwright, "Phase Transition for a Hard Sphere System", *J. Chem. Phys.* **27**, 1208–1209 (1957).
- [22] A. Rahman, "Correlations in the Motion of Atoms in Liquid Argon", *Phys. Rev.* **136**, A405–A411 (1964).
- [23] J. Liouville, "Note sur la Théorie de la Variation des constantes arbitraires", *J Math Pures Appl*, 342–349 (1838).
- [24] J. W. Gibbs, "On the Fundamental Formula of Statistical Mechanics, with Applications to Astronomy and Thermodynamics", *Proc. Am. Assoc. Adv. Sci.* **33**, 57–58 (1884).
- [25] J. W. Gibbs, *Elementary Principles in Statistical Mechanics, developed with especial reference to the rational foundation of thermodynamics* (Charles Scribner's Sons, 1902).
- [26] W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson, "A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters", *J. Chem. Phys.* **76**, 637–649 (1982).
- [27] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, "Molecular dynamics with coupling to an external bath", *J. Chem. Phys.* **81**, 3684–3690 (1984).

-
- [28] P. Langevin, “Sur la théorie du mouvement brownien”, *C. R. Acad. Sci.* **146**, 530–533 (1908).
- [29] D. S. Lemons and A. Gythiel, “Paul Langevin’s 1908 paper “On the Theory of Brownian Motion” [“Sur la théorie du mouvement brownien”, *C. R. Acad. Sci. (Paris)* 146, 530–533 (1908)]”, *Am. J. Phys.* **65**, 1079–1081 (1997).
- [30] H. Mori, “Transport, Collective Motion, and Brownian Motion”, *Prog. Theor. Phys.* **33**, 423–455 (1965).
- [31] R Kubo, “The fluctuation-dissipation theorem”, *Rep. Prog. Phys.* **29**, 255–284 (1966).
- [32] G. Jung, M. Hanke, and F. Schmid, “Iterative Reconstruction of Memory Kernels”, *J. Chem. Theory Comput.* **13**, 2481–2488 (2017).
- [33] N Bockius, J Shea, G Jung, F Schmid, and M Hanke, “Model reduction techniques for the computation of extended Markov parameterizations for generalized Langevin equations”, *J. Phys. Condens. Matter* **33**, 214003 (2021).
- [34] M. S. Daw and M. I. Baskes, “Semiempirical, Quantum Mechanical Calculation of Hydrogen Embrittlement in Metals”, *Phys. Rev. Lett.* **50**, 1285–1288 (1983).
- [35] M. S. Daw and M. I. Baskes, “Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals”, *Phys. Rev. B* **29**, 6443–6453 (1984).
- [36] M. S. Shell, “The relative entropy is fundamental to multiscale and inverse thermodynamic problems”, *J. Chem. Phys.* **129**, 144108 (2008).
- [37] T. Sanyal and M. S. Shell, “Coarse-grained models using local-density potentials optimized with the relative entropy: Application to implicit solvation”, *J. Chem. Phys.* **145**, 034109 (2016).
- [38] J. D. van der Waals, “The law of corresponding states for different substances”, in *KNAW Proceedings* (1913), pp. 971–981.
- [39] M. G. Noro and D. Frenkel, “Extended corresponding-states behavior for particles with variable range attractions”, *J. Chem. Phys.* **113**, 2941–2944 (2000).
- [40] D. Frenkel and B. Smit, *Understanding Molecular Simulation* (Elsevier, 2002).
- [41] C. Yuan, S. Li, Q. Zou, Y. Ren, and X. Yan, “Multiscale simulations for understanding the evolution and mechanism of hierarchical peptide self-assembly”, *Phys. Chem. Chem. Phys.* **19**, 23614–23631 (2017).

- [42] J. F. Rudzinski and W. G. Noid, “Coarse-graining entropy, forces, and structures”, *J. Chem. Phys.* **135**, 214101 (2011).
- [43] A. Soper, “Empirical potential Monte Carlo simulation of fluid structure”, *Chem. Phys.* **202**, 295–306 (1996).
- [44] D. Reith, M. Pütz, and F. Müller-Plathe, “Deriving effective mesoscale potentials from atomistic simulations, Mesoscale Potentials from Atomistic Simulations”, *J. Comput. Chem.* **24**, 1624–1636 (2003).
- [45] R. Henderson, “A uniqueness theorem for fluid pair correlation functions”, *Phys. Lett. A* **49**, 197–198 (1974).
- [46] A. P. Lyubartsev and A. Laaksonen, “Calculation of effective interaction potentials from radial distribution functions: A reverse Monte Carlo approach”, *Phys. Rev. E* **52**, 3730–3737 (1995).
- [47] F. Delbary, M. Hanke, and D. Ivanizki, “A generalized Newton iteration for computing the solution of the inverse Henderson problem”, *Inverse Probl. Sci. En.* **28**, 1166–1190 (2020).
- [48] M. P. Bernhardt, M. Hanke, and N. F. A. van der Vegt, “Iterative integral equation methods for structural coarse-graining”, *J. Chem. Phys.* **154**, 084118 (2021).
- [49] F. Ercolessi and J. B. Adams, “Interatomic Potentials from First-Principles Calculations: The Force-Matching Method”, *Europhys. Lett.* **26**, 583–588 (1994).
- [50] S. Izvekov and G. A. Voth, “Multiscale coarse graining of liquid-state systems”, *J. Chem. Phys.* **123**, 134105 (2005).
- [51] W. G. Noid, J.-W. Chu, G. S. Ayton, and G. A. Voth, “Multiscale Coarse-Graining and Structural Correlations: Connections to Liquid-State Theory”, *J. Phys. Chem. B* **111**, 4116–4127 (2007).
- [52] A. Chaimovich and M. S. Shell, “Coarse-graining errors and numerical optimization using a relative entropy framework”, *J. Chem. Phys.* **134**, 094112 (2011).
- [53] S. Kullback and R. A. Leibler, “On Information and Sufficiency”, *Ann. Math. Statist.* **22**, 79–86 (1951).
- [54] D. Wu and D. A. Kofke, “Phase-space overlap measures. II. Design and implementation of staging methods for free-energy calculations”, *J. Chem. Phys.* **123**, 084109 (2005).
- [55] C. de Boor, *A practical guide to splines* (Springer, 1978).
- [56] T. M. Mitchell, *Machine Learning* (McGraw-Hill, 1997).

-
- [57] I. J. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
- [58] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (A Bradford Book, 2018).
- [59] D. Hebb, *The Organization of Behavior* (Psychology Press, 2005).
- [60] S. Löwel and W. Singer, “Selection of Intrinsic Horizontal Connections in the Visual Cortex by Correlated Neuronal Activity”, *Science* **255**, 209–212 (1992).
- [61] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory”, *Neural Comput.* **9**, 1735–1780 (1997).
- [62] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is All you Need”, in *Advances in Neural Information Processing Systems*, Vol. 30, edited by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (2017).
- [63] L. M. Rios and N. V. Sahinidis, “Derivative-free optimization: A review of algorithms and comparison of software implementations”, *J. Global Optim.* **56**, 1247–1293 (2012).
- [64] S. Dreyfus, “The computational solution of optimal control problems with time lag”, *IEEE Trans. Autom. Control* **18**, 383–385 (1973).
- [65] P. J. Werbos, *The roots of backpropagation: From ordered derivatives to neural networks and political forecasting* (Wiley, Nashville, TN, 1994).
- [66] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors”, *Nature* **323**, 533–536 (1986).
- [67] D. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization”, *arXiv* (2014).
- [68] F. Berressem and A. Nikoubashman, “BoltzmaNN: Predicting effective pair potentials and equations of state using neural networks”, *J. Chem. Phys.* **154**, 124123 (2021).
- [69] L. Allen, A. O’Connell, and V. Kiermer, “How can we ensure visibility and diversity in research contributions? How the Contributor Role Taxonomy (CRediT) is helping the shift from authorship to contributorship”, *Learn. Publ.* **32**, 71–74 (2019).
- [70] P. J. Lu, E. Zaccarelli, F. Ciulla, A. B. Schofield, F. Sciortino, and D. A. Weitz, “Gelation of particles with short-range attraction”, *Nature* **453**, 499–503 (2008).

- [71] P. G. Bolhuis, A. A. Louis, J. P. Hansen, and E. J. Meijer, “Accurate effective pair potentials for polymer solutions”, *J. Chem. Phys.* **114**, 4296–4311 (2001).
- [72] C. N. Likos, “Effective interactions in soft condensed matter physics”, *Phys. Rep.* **348**, 267–439 (2001).
- [73] F. Sciortino, S. Mossa, E. Zaccarelli, and P. Tartaglia, “Equilibrium Cluster Phases and Low-Density Arrested Disordered States: The Role of Short-Range Attraction and Long-Range Repulsion”, *Phys. Rev. Lett.* **93**, 055701 (2004).
- [74] S. Mossa, F. Sciortino, P. Tartaglia, and E. Zaccarelli, “Ground-State Clusters for Short-Range Attractive and Long-Range Repulsive Potentials”, *Langmuir* **20**, 10756–10763 (2004).
- [75] B. M. Mladek, G. Kahl, and C. N. Likos, “Computer Assembly of Cluster-Forming Amphiphilic Dendrimers”, *Phys. Rev. Lett.* **100**, 028301 (2008).
- [76] A. Jain, J. A. Bollinger, and T. M. Truskett, “Inverse methods for material design”, *AIChE J.* **60**, 2732–2740 (2014).
- [77] Z. M. Sherman, M. P. Howard, B. A. Lindquist, R. B. Jadrich, and T. M. Truskett, “Inverse methods for design of soft materials”, *J. Chem. Phys.* **152**, 140902 (2020).
- [78] H. M. Jaeger, “Celebrating Soft Matter’s 10th Anniversary: Toward jamming by design”, *Soft Matter* **11**, 12–27 (2015).
- [79] B. A. Lindquist, R. B. Jadrich, and T. M. Truskett, “Communication: Inverse design for self-assembly via on-the-fly optimization”, *J. Chem. Phys.* **145**, 111101 (2016).
- [80] D. Chen, G. Zhang, and S. Torquato, “Inverse Design of Colloidal Crystals via Optimized Patchy Interactions”, *J. Phys. Chem. B* **122**, 8462–8468 (2018).
- [81] M. R. Khadilkar, S. Paradiso, K. T. Delaney, and G. H. Fredrickson, “Inverse Design of Bulk Morphologies in Multiblock Polymers Using Particle Swarm Optimization”, *Macromolecules* **50**, 6702–6709 (2017).
- [82] K. R. Gadelrab, A. F. Hannon, C. A. Ross, and A. Alexander-Katz, “Inverting the design path for self-assembled block copolymers”, *Mol. Syst. Des. Eng.* **2**, 539–548 (2017).
- [83] J. Behler, “Perspective: Machine learning potentials for atomistic simulations”, *J. Chem. Phys.* **145**, 170901 (2016).
- [84] D. J. Audus and J. J. de Pablo, “Polymer Informatics: Opportunities and Challenges”, *ACS Macro Lett.* **6**, 1078–1082 (2017).

- [85] A. L. Ferguson, “Machine learning and data science in soft materials engineering”, *J. Phys. Condens. Matter* **30**, 043002 (2017).
- [86] T. Bereau, R. A. DiStasio, A. Tkatchenko, and O. A. von Lilienfeld, “Non-covalent interactions across organic and biological subsets of chemical space: Physics-based potentials parametrized from machine learning”, *J. Chem. Phys.* **148**, 241706 (2018).
- [87] T. Bereau, “Data-Driven Methods in Multiscale Modeling of Soft Matter”, in *Handbook of Materials Modeling*, edited by W. Andreoni and S. Yip (Springer International Publishing, Cham, 2018), pp. 1–12.
- [88] N. E. Jackson, M. A. Webb, and J. J. de Pablo, “Recent advances in machine learning towards multiscale soft materials design”, *Curr. Opin. Chem. Eng.* **23**, 106–114 (2019).
- [89] S. Wu, Y. Kondo, M.-a. Kakimoto, B. Yang, H. Yamada, I. Kuwajima, G. Lambard, K. Hongo, Y. Xu, J. Shiomi, C. Schick, J. Morikawa, and R. Yoshida, “Machine-learning-assisted discovery of polymers with high thermal conductivity using a molecular design algorithm”, *npj Comput. Mater.* **5**, 66 (2019).
- [90] J. Schmidt, M. R. G. Marques, S. Botti, and M. A. L. Marques, “Recent advances and applications of machine learning in solid-state materials science”, *npj Comput. Mater.* **5**, 83 (2019).
- [91] R. L. McGreevy and L. Pusztai, “Reverse Monte Carlo Simulation: A New Technique for the Determination of Disordered Structures”, *Mol. Simulat.* **1**, 359–367 (1988).
- [92] D. A. Keen and R. L. McGreevy, “Structural modelling of glasses using reverse Monte Carlo simulation”, *Nature* **344**, 423–425 (1990).
- [93] D. Reith, M. Pütz, and F. Müller-Plathe, “Deriving effective mesoscale potentials from atomistic simulations”, *J. Comput. Chem.* **24**, 1624–1636 (2003).
- [94] M. C. Rechtsman, F. H. Stillinger, and S. Torquato, “Optimized Interactions for Targeted Self-Assembly: Application to a Honeycomb Lattice”, *Phys. Rev. Lett.* **95**, 228301 (2005).
- [95] L. Reatto, D. Levesque, and J. J. Weis, “Iterative predictor-corrector method for extraction of the pair interaction from structural data for dense classical liquids”, *Phys. Rev. A* **33**, 3451–3465 (1986).

- [96] K. Müller, N. Osterman, D. Babič, C. N. Likos, J. Dobnikar, and A. Nikoubashman, “Pattern Formation and Coarse-Graining in Two-Dimensional Colloids Driven by Multiaxial Magnetic Fields”, *Langmuir* **30**, 5088–5096 (2014).
- [97] L. Song, L. Yang, J. Meng, and S. Yang, “Thermodynamics of Hydrophobic Amino Acids in Solution: A Combined Experimental–Computational Study”, *J. Phys. Chem. Lett.* **8**, 347–351 (2017).
- [98] F. Müller-Plathe, “Coarse-Graining in Polymer Simulation: From the Atomistic to the Mesoscopic Scale and Back”, *ChemPhysChem* **3**, 754–769 (2002).
- [99] G. Milano and F. Müller-Plathe, “Mapping Atomistic Simulations to Mesoscopic Models: A Systematic Coarse-Graining Procedure for Vinyl Polymer Chains”, *J. Phys. Chem. B* **109**, 18609–18619 (2005).
- [100] B. Bayramoglu and R. Faller, “Coarse-Grained Modeling of Polystyrene in Various Environments by Iterative Boltzmann Inversion”, *Macromolecules* **45**, 9205–9219 (2012).
- [101] A. A. Louis, “Beware of density dependent pair potentials”, *J. Phys. Condens. Matter* **14**, 9187–9206 (2002).
- [102] D. Rosenberger, M. Hanke, and N. F. van der Vegt, “Comparison of iterative inverse coarse-graining methods”, *Eur. Phys. J. Special Topics* **225**, 1323–1345 (2016).
- [103] H. Wang, C. Junghans, and K. Kremer, “Comparative atomistic and coarse-grained study of water: What do we lose by coarse-graining?”, *Eur. Phys. J. E* **28**, 221–229 (2009).
- [104] A. Das and H. C. Andersen, “The multiscale coarse-graining method. V. Isothermal-isobaric ensemble”, *J. Chem. Phys.* **132**, 164106 (2010).
- [105] T. Bereau and J. F. Rudzinski, “Accurate Structure-Based Coarse Graining Leads to Consistent Barrier-Crossing Dynamics”, *Phys. Rev. Lett.* **121**, 256002 (2018).
- [106] K. M. Lebold and W. G. Noid, “Dual approach for effective potentials that accurately model structure and energetics”, *J. Chem. Phys.* **150**, 234107 (2019).
- [107] C. Scherer, R. Scheid, D. Andrienko, and T. Bereau, “Kernel-Based Machine Learning for Efficient Simulations of Molecular Liquids”, *J. Chem. Theory Comput.* **16**, 3194–3204 (2020).

- [108] M. K. Meinel and F. Müller-Plathe, “Loss of Molecular Roughness upon Coarse-Graining Predicts the Artificially Accelerated Mobility of Coarse-Grained Molecular Simulation Models”, *J. Chem. Theory Comput.* **16**, 1411–1419 (2020).
- [109] T. T. Foley, K. M. Kidder, M. S. Shell, and W. G. Noid, “Exploring the landscape of model representations”, *Proc. Natl. Acad. Sci.* **117**, 24061–24068 (2020).
- [110] See <https://gitlab.rlp.net/anikouba/boltzmann> for the source code and web-based implementations of our programs.
- [111] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation”, in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, edited by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi (2015), pp. 234–241.
- [112] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: Large-scale Machine Learning on Heterogeneous Systems*, Software available from tensorflow.org, 2015.
- [113] J. A. Anderson, J. Glaser, and S. C. Glotzer, “HOOMD-blue: A Python package for high-performance molecular dynamics and hard particle Monte Carlo simulations”, *Comput. Mater. Sci.* **173**, 109363 (2020).
- [114] J.-P. Hansen and I. R. McDonald, *Theory of Simple Liquids*, 3rd ed. (Academic Press, 2006).
- [115] V. Rühle, C. Junghans, A. Lukyanov, K. Kremer, and D. Andrienko, “Versatile Object-Oriented Toolkit for Coarse-Graining Applications”, *J. Chem. Theory Comput.* **5**, 3211–3223 (2009).
- [116] J. J. Potoff and A. Z. Panagiotopoulos, “Critical point and phase behavior of the pure fluid and a Lennard-Jones mixture”, *J. Chem. Phys.* **109**, 10914–10920 (1998).
- [117] J. K. Johnson, J. A. Zollweg, and K. E. Gubbins, “The Lennard-Jones equation of state revisited”, *Mol. Phys.* **78**, 591–618 (1993).
- [118] N. F. Carnahan and K. E. Starling, “Equation of State for Nonattracting Rigid Spheres”, *J. Chem. Phys.* **51**, 635–636 (1969).

- [119] C.-C. Fu, P. M. Kulkarni, M. Scott Shell, and L. Gary Leal, “A test of systematic coarse-graining of molecular dynamics simulations: Thermodynamic properties”, *J. Chem. Phys.* **137**, 164106 (2012).
- [120] N. J. H. Dunn and W. G. Noid, “Bottom-up coarse-grained models that accurately describe the structure, pressure, and compressibility of molecular liquids”, *J. Chem. Phys.* **143**, 243148 (2015).
- [121] B. A. Lindquist, R. B. Jadrich, M. P. Howard, and T. M. Truskett, “The role of pressure in inverse design for assembly”, *J. Chem. Phys.* **151**, 104104 (2019).
- [122] F. Berressem, C. Scherer, D. Andrienko, and A. Nikoubashman, “Ultra-coarse-graining of homopolymers in inhomogeneous systems”, *J. Phys. Condens. Matter* **33**, 254002 (2021).
- [123] A. A. Louis, P. G. Bolhuis, J. P. Hansen, and E. J. Meijer, “Can Polymer Coils Be Modeled as “Soft Colloids”?”, *Phys. Rev. Lett.* **85**, 2522–2525 (2000).
- [124] G. Zhang, A. Chazirakis, V. A. Harmandaris, T. Stuehn, K. C. Daoulas, and K. Kremer, “Hierarchical modelling of polystyrene melts: From soft blobs to atomistic resolution”, *Soft Matter* **15**, 289–302 (2019).
- [125] N. Zacharopoulos, N. Vergadou, and D. N. Theodorou, “Coarse graining using pretabulated potentials: Liquid benzene”, *J. Chem. Phys.* **122**, 244111 (2005).
- [126] J. W. Mullinax and W. G. Noid, “A Generalized-Yvon–Born–Green Theory for Determining Coarse-Grained Interaction Potentials”, *J. Phys. Chem. C* **114**, 5661–5674 (2009).
- [127] J. W. Mullinax and W. G. Noid, “Reference state for the generalized Yvon–Born–Green theory: Application for coarse-grained model of hydrophobic hydration”, *J. Chem. Phys.* **133**, 124107 (2010).
- [128] T. Murtola, A. Bunker, I. Vattulainen, M. Deserno, and M. Karttunen, “Multiscale modeling of emergent materials: Biological and soft matter”, *Phys. Chem. Chem. Phys.* **11**, 1869 (2009).
- [129] S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. de Vries, “The MARTINI Force Field: Coarse Grained Model for Biomolecular Simulations”, *J. Phys. Chem. B* **111**, 7812–7824 (2007).
- [130] D. M. Huang, R. Faller, K. Do, and A. J. Moulé, “Coarse-Grained Computer Simulations of Polymer/Fullerene Bulk Heterojunctions for Organic Photovoltaic Applications”, *J. Chem. Theory Comput.* **6**, 526–537 (2009).

- [131] A. Lukyanov, A. Malafeev, V. Ivanov, H.-L. Chen, K. Kremer, and D. Andrienko, "Solvated poly-(phenylene vinylene) derivatives: Conformational structure and aggregation behavior", *J. Mater. Chem.* **20**, 10475 (2010).
- [132] V. Rühle, J. Kirkpatrick, K. Kremer, and D. Andrienko, "Coarse-grained modelling of polypyrrole morphologies", *Phys. Status Solidi B* **245**, 844–848 (2008).
- [133] P. Gemünden, C. Poelking, K. Kremer, D. Andrienko, and K. C. Daoulas, "Nematic Ordering, Conjugation, and Density of States of Soluble Polymeric Semiconductors", *Macromolecules* **46**, 5762–5774 (2013).
- [134] C. Scherer and D. Andrienko, "Comparison of systematic coarse-graining strategies for soluble conjugated polymers", *Eur. Phys. J. Special Topics* **225**, 1441–1461 (2016).
- [135] C. Greco, A. Melnyk, K. Kremer, D. Andrienko, and K. C. Daoulas, "Generic Model for Lamellar Self-Assembly in Conjugated Polymers: Linking Mesoscopic Morphology and Charge Transport in P3HT", *Macromolecules* **52**, 968–981 (2019).
- [136] A. Villa, C. Peter, and N. F. A. van der Vegt, "Self-assembling dipeptides: Conformational sampling in solvent-free coarse-grained simulation", *Phys. Chem. Chem. Phys.* **11**, 2077 (2009).
- [137] W. Shinoda, R. DeVane, and M. L. Klein, "Coarse-grained molecular modeling of non-ionic surfactant self-assembly", *Soft Matter* **4**, 2454 (2008).
- [138] L. Monticelli, S. K. Kandasamy, X. Periole, R. G. Larson, D. P. Tieleman, and S.-J. Marrink, "The MARTINI Coarse-Grained Force Field: Extension to Proteins", *J. Chem. Theory Comput.* **4**, 819–834 (2008).
- [139] W. G. Noid, J.-W. Chu, G. S. Ayton, V. Krishna, S. Izvekov, G. A. Voth, A. Das, and H. C. Andersen, "The multiscale coarse-graining method. I. A rigorous bridge between atomistic and coarse-grained models", *J. Chem. Phys.* **128**, 244114 (2008).
- [140] S. Y. Mashayak, M. N. Jochum, K. Koschke, N. R. Aluru, V. Rühle, and C. Junghans, "Relative Entropy and Optimization-Driven Coarse-Graining Methods in VOTCA", *PLoS One* **10**, e0131754 (2015).
- [141] E. Kalligiannaki, A. Chazirakis, A. Tsourtis, M. Katsoulakis, P. Plecháč, and V. Harmandaris, "Parametrizing coarse grained models for molecular systems at equilibrium", *Eur. Phys. J. Special Topics* **225**, 1347–1372 (2016).

- [142] W. G. Noid, P. Liu, Y. Wang, J.-W. Chu, G. S. Ayton, S. Izvekov, H. C. Andersen, and G. A. Voth, “The multiscale coarse-graining method. II. Numerical implementation for coarse-grained molecular models”, *J. Chem. Phys.* **128**, 244115 (2008).
- [143] G. Tóth, “Effective potentials from complex simulations: A potential-matching algorithm and remarks on coarse-grained potentials”, *J. Phys. Condens. Matter* **19**, 335222 (2007).
- [144] C. Scherer and D. Andrienko, “Understanding three-body contributions to coarse-grained force fields”, *Phys. Chem. Chem. Phys.* **20**, 22387–22394 (2018).
- [145] C. Peter and K. Kremer, “Multiscale simulation of soft matter systems – from the atomistic to the coarse-grained level and back”, *Soft Matter* **5**, 4357 (2009).
- [146] T. Bereau, W. F. D. Bennett, J. Pfaendtner, M. Deserno, and M. Karttunen, “Folding and insertion thermodynamics of the transmembrane WALP peptide”, *J. Chem. Phys.* **143**, 243127 (2015).
- [147] M. Kanduč and R. R. Netz, “Atomistic simulations of wetting properties and water films on hydrophilic surfaces”, *J. Chem. Phys.* **146**, 164705 (2017).
- [148] K. Binder, “Critical Properties from Monte Carlo Coarse Graining and Renormalization”, *Phys. Rev. Lett.* **47**, 693–696 (1981).
- [149] C. Svaneborg, H. A. Karimi-Varzaneh, N. Hojdis, F. Fleck, and R. Everaers, “Multiscale approach to equilibrating model polymer melts”, *Phys. Rev. E* **94**, 032502 (2016).
- [150] J. Jin and G. A. Voth, “Ultra-Coarse-Grained Models Allow for an Accurate and Transferable Treatment of Interfacial Systems”, *J. Chem. Theory Comput.* **14**, 2180–2197 (2018).
- [151] M. Jochum, D. Andrienko, K. Kremer, and C. Peter, “Structure-based coarse-graining in liquid slabs”, *J. Chem. Phys.* **137**, 064102 (2012).
- [152] S. Y. Mashayak and N. R. Aluru, “Coarse-Grained Potential Model for Structural Prediction of Confined Water”, *J. Chem. Theory Comput.* **8**, 1828–1840 (2012).
- [153] M. Montes-Saralegui, G. Kahl, and A. Nikoubashman, “On the applicability of density dependent effective interactions in cluster-forming systems”, *J. Chem. Phys.* **146**, 054904 (2017).

- [154] A. Das and H. C. Andersen, “The multiscale coarse-graining method. IX. A general method for construction of three body coarse-grained force fields”, *J. Chem. Phys.* **136**, 194114 (2012).
- [155] T. Bereau, D. Andrienko, and O. A. von Lilienfeld, “Transferable Atomic Multipole Machine Learning Models for Small Organic Molecules”, *J. Chem. Theory Comput.* **11**, 3225–3233 (2015).
- [156] A. V. Shapeev, “Moment Tensor Potentials: A Class of Systematically Improvable Interatomic Potentials”, *MMS* **14**, 1153–1173 (2016).
- [157] T. T. Nguyen, E. Székely, G. Imbalzano, J. Behler, G. Csányi, M. Ceriotti, A. W. Götz, and F. Paesani, “Comparison of permutationally invariant polynomials, neural networks, and Gaussian approximation potentials in representing water interactions through many-body expansions”, *J. Chem. Phys.* **148**, 241725 (2018).
- [158] A. Glielmo, C. Zeni, and A. De Vita, “Efficient nonparametric n -body force fields from machine learning”, *Phys. Rev. B* **97**, 184307 (2018).
- [159] E. C. Allen and G. C. Rutledge, “A novel algorithm for creating coarse-grained, density dependent implicit solvent models”, *J. Chem. Phys.* **128**, 154115 (2008).
- [160] G. Faure, J.-B. Maillet, and G. Stoltz, “Local density dependent potential for compressible mesoparticles”, *J. Chem. Phys.* **140**, 114105 (2014).
- [161] M. R. DeLyster and W. G. Noid, “Extending pressure-matching to inhomogeneous systems via local-density potentials”, *J. Chem. Phys.* **147**, 134111 (2017).
- [162] M. R. DeLyster and W. G. Noid, “Analysis of local density potentials”, *J. Chem. Phys.* **151**, 224106 (2019).
- [163] J. Midya, S. A. Egorov, K. Binder, and A. Nikoubashman, “Phase behavior of flexible and semiflexible polymers in solvents of varying quality”, *J. Chem. Phys.* **151**, 034902 (2019).
- [164] W. Humphrey, A. Dalke, and K. Schulten, “VMD: Visual molecular dynamics”, *J. Mol. Graphics* **14**, 33–38 (1996).
- [165] K. Binder, M. Müller, F. Schmid, and A. Werner, “‘Intrinsic’ profiles and capillary waves at interfaces between coexisting phases in polymer blends”, *Adv. Colloid Interfac.* **94**, 237–248 (2001).
- [166] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, “GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers”, *SoftwareX* **1-2**, 19–25 (2015).

- [167] S. Plimpton, “Fast Parallel Algorithms for Short-Range Molecular Dynamics”, *J. Comput. Phys.* **117**, 1–19 (1995).
- [168] H. J. C. Berendsen, J. R. Grigera, and T. P. Straatsma, “The missing term in effective pair potentials”, *J Phys Chem* **91**, 6269–6271 (1987).
- [169] I. Morozov, A. Kazennov, R. Bystryi, G. Norman, V. Pisarev, and V. Stegailov, “Molecular dynamics simulations of the relaxation processes in the condensed matter on GPUs”, *Comput. Phys. Commun.* **182**, 1974–1978 (2011).
- [170] L. Yang, F. Zhang, C.-Z. Wang, K.-M. Ho, and A. Travesset, “Implementation of metal-friendly EAM/FS-type semi-empirical potentials in HOOMD-blue: A GPU-accelerated molecular dynamics software”, *J. Comput. Phys.* **359**, 352–360 (2018).
- [171] P. J. Flory, *Statistical mechanics of chain molecules* (Oxford University Press, 1988).
- [172] J. P Wittmer, P Beckrich, A Johner, A. N Semenov, S. P Obukhov, H Meyer, and J Baschnagel, “Why polymer chains in a melt are not random walks”, *Europhys. Lett.* **77**, 56003 (2007).
- [173] H. Meyer, E. Horwath, and P. Virnau, “Mapping onto Ideal Chains Overestimates Self-Entanglements in Polymer Melts”, *ACS Macro Lett.* **7**, 757–761 (2018).
- [174] H. Meyer, J. P. Wittmer, T. Kreer, P. Beckrich, A. Johner, J. Farago, and J. Baschnagel, “Static Rouse modes and related quantities: Corrections to chain ideality in polymer melts”, *Eur. Phys. J. E* **26**, 25–33 (2008).
- [175] B. Kruger, L. Schäfer, and A. Baumgärtner, “Correlations among interpenetrating polymer coils : The probing of a fractal”, *J. Phys.* **50**, 3191–3222 (1989).
- [176] O. F. Olaj and K. H. Pelinka, “Pair distribution function and pair potential of lattice model chains under theta conditions, 1. Numerical evaluation”, *Makromol. Chem.* **177**, 3413–3425 (1976).
- [177] G. Reddy and A. Yethiraj, “Implicit and Explicit Solvent Models for the Simulation of Dilute Polymer Solutions”, *Macromolecules* **39**, 8536–8542 (2006).
- [178] S. Huissmann, R. Blaak, and C. N. Likos, “Star Polymers in Solvents of Varying Quality”, *Macromolecules* **42**, 2806–2816 (2009).
- [179] M. G. Martin and J. I. Siepmann, “Transferable Potentials for Phase Equilibria. 1. United-Atom Description of *n*-Alkanes”, *J. Phys. Chem. B* **102**, 2569–2577 (1998).

-
- [180] H. A. Lorentz, "Ueber die Anwendung des Satzes vom Virial in der kinetischen Theorie der Gase", *Ann. Phys.* **248**, 127–136 (1881).
- [181] D. Berthelot, "Sur le Mélange des Gaz", *C. R. Acad. Sci.* **126**, 1703–1706 (1898).
- [182] W. L. Jorgensen and J. Tirado-Rives, "The OPLS [optimized potentials for liquid simulations] potential functions for proteins, energy minimizations for crystals of cyclic peptides and crambin", *JACS* **110**, 1657–1666 (1988).
- [183] R. M. Patel, "Polyethylene", in *Multilayer Flexible Packaging (Second Edition)*, Second Edition, Plastics Design Library (William Andrew Publishing, 2016), pp. 17–34.
- [184] H. Yamakawa, *Modern theory of polymer solutions* (Harper & Row, 1971).
- [185] P. Debye and F. Bueche, "Distribution of Segments in a Coiling Polymer Molecule", *J. Chem. Phys.* **20**, 1337–1338 (1952).

Acknowledgments

In the last few years, I have spent a lot of time researching and writing this thesis. While this has been an enlightening and pleasant experience, it has also been a demanding one of course. During my journey I have met new people, made new friends and received a lot of support, for which I am very grateful. To all those accompanying me during this time, I would like to express my gratitude. Especially I would like to thank the following people:

First of all, Arash Nikoubashman for giving me the opportunity to work on this interesting topic and for being a mentor to me both scientifically as well as personal for more than five years. I am grateful to have had such an ambitious but also patient supervisor and I wish you all the best.

Second, to Friederike Schmid for many interesting discussions and insights as well as for always providing new perspectives on my research.

The whole group in Mainz and Darmstadt for all the scientific discussions and the fun we had together. Especially I want to thank my colleagues from my office, Yannick Witzky, Kyra Klos, Simon Lemcke and Le Qiao as well as Jörn Appeldorn, Jeanine Shea, Oliver Habrich, Christoph Scherer and Janka Bauer to name but a few. Furthermore, I would like to thank Andreas Nussbaumer for all his technical support and Daniela Reibel, Markus Haack and Mariana Cosarinsky for all the administrative support.

All my friends both in as well as outside of university for accompanying me and enriching my life. I am especially grateful to all my fellow students for always sticking together and supporting each other.

My family for always having my back and believing in me. In particular, I want to thank my grandparents, my parents and my brother for all the encouragement and all the love. Without you, my studies and this thesis would not have been possible.

Finally, I would like to thank Alina Kirschen for all the love, care and support. I am grateful for all the great memories we share and all the good and bad times we have endured without ever doubting our love. For the last ten years you have

Appendix D. Acknowledgments

made me incredibly happy and I am looking forward to a bright future with you.

Curriculum Vitae

Personal Information

Name: Fabian Berressem, M.Sc
Marital Status: Single
Nationality: German
Date of Birth: January 27, 1994
Place of Birth: Lahnstein
Current Address: Roonstrasse 10
35390 Giessen
Mail Address: fberres@uni-mainz.de



Work Experience

| | | |
|-----------------------|--|-----------------|
| Aug. 2022 - present | Data Scientist (Enterprise Advanced Analytics) | 1&1 Telecom. SE |
| Nov. 2018 - Jun. 2022 | Research assistant in Physics | JGU Mainz |
| Oct. 2015 - Sep. 2017 | Student assistant in Physics | JGU Mainz |

Education

| | | |
|-----------------------|---|-----------|
| Nov. 2018 - present | PhD Thesis on “Coarse-graining and inverse design in soft matter via local density-dependent potentials and machine learning methods” | JGU Mainz |
| Sep. 2017 - Oct. 2018 | Master thesis on “Static and Dynamic Properties of Diblock and Multiblock Copolymers” | JGU Mainz |
| Oct. 2016 - Oct. 2018 | Graduate Studies in Physics with Degree “M.Sc” Grade: 1.0 \cong A+, <i>Predicate: “with distinction”</i> | JGU Mainz |
| Oct. 2013 - Oct. 2016 | Undergraduate Studies in Physics with Degree “B.Sc” Grade: 1.8 \cong A- | JGU Mainz |