Molecular Phenomena in Dynamic Wetting: Superspreading and Precursors

Von der Fakultät für Maschinenwesen der
Rheinisch-Westfälischen Technischen Hochschule Aachen zur
Erlangung des akademischen Grades eines Doktors der
Ingenieurwissenschaften genehmigte Dissertation

vorgelegt von

Rolf Erwin Isele-Holder

Berichter: Juniorprofessor Ahmed E. Ismail, Ph. D.
Juniorprofessor Dr.-Ing. Martin Thomas Horsch

Tag der mündlichen Prüfung: 17.04.2015

Diese Dissertation ist auf den Internetseiten der Universitätsbibliothek online verfügbar.

Abstract

Wetting is a multiscale process that can be controlled simultaneously by complex flow patterns on the macroscale and contact line phenomena at the Ångstrom scale. While resolving the latter scale is often circumvented by usage of boundary conditions, there are molecular wetting phenomena in which this approach is infeasible. The focus of this study is to use molecular dynamics simulations to examine two of these phenomena: superspreading, the ultra-rapid wetting of aqueous solutions facilitated by trisiloxane surfactants, and molecular precursors, the development of films of molecular thickness that precede droplets.

Molecular simulation resolves the atomistic scale and provides information that is inaccessible from experiment. A challenge in the context of wetting, however, is that dispersion interactions are typically considered short-ranged in molecular simulations, whereas they have long-ranged effects in wetting. To capture these interactions in wetting simulations, the particle-particle particle-mesh algorithm, a long-range solver that is well-established for Coulomb interactions, is extended to dispersion. It is shown that the correct use of this algorithm leads to accurate and efficient simulations.

Despite intensive studies on superspreading in the last 20 years, the underlying molecular mechanisms of the process are not understood. That the process is sensitive to various parameters in experiment, and also that previous attempts to model this phenomenon using molecular dynamics simulations failed, motivated the development of a force field dedicated to superspreading. Application in large-scale spreading simulations provides a smooth contact line transition at superspreading conditions. It is shown that this observation offers plausible explanations for experimental findings and a coherent description of the superspreading mechanism.

While the dynamics and mass transport mechanisms of molecular precursors are well understood, conditions that lead to precursor formation or different types of precursors are subject to debate. Large-scale spreading simulations, new analysis methods, and excessive free energy computations shed light on these issues and resolve the conflict about the role of the spreading coefficient for precursor formation.

Zusammenfassung

Benetzung ist ein vielskaliger Prozess der simultan von komplexen Strömungsbewegungen auf der makroskopischen Skala und Kontaklinienphänomenen auf der Ångstrom-Skala kontrolliert werden kann. Obwohl die Modellierung der letzgenannten Skalen oft durch geeignete Randbedingungen umgangen wird, gibt es molekulare Benetzungsphänomene, in denen diese Herangehensweise nicht möglich ist. Der Fokus dieser Studie ist die Untersuchung zwei dieser Phänomene mit molekulardynamischen Simulationen: Superspreading, das durch Trisiloxantenside ermöglichte ultra-schnelle Benetzen von hydrophoben Substraten durch wässrige Lösungen, und molekulare Präkursoren, die Entwicklung von Filmen molekularer Dicke, die Tropfen voraus fließen.

Molekulare Simulation löst atomistische Skalen auf und liefert Informationen, die experimentell nicht zugänglich sind. Eine Herausvorderung im Kontext von Benetzung ist jedoch, dass Dispersionswechselwirkungen in molekularen Simulationen typischerweise als kurzreichweitig angesehen werden, wohingegen sie in Benetzung langreichweitige Effekte haben. Um diese Wechselwirkungen in Simulationen von Benetzung zu berücksichtigen, wurde der particle-particle particle-mesh Algorithmus, ein für Coulomb Wechselwirkungen weit verbreiteter langreichweitiger Löser, für Dispersionswechselwirkungen erweitert. Es wird gezeigt, dass die korrekte Verwendung dieses Algorithmus genaue und effiziente Berechnungen ermöglicht.

Trotz intensiven Studien zu Superspreading in den vergangenen 20 Jahren sind die zu Grunde liegenden molekularen Mechanismen nicht verstanden. Dass der Prozess empfindlich gegenüber experimentellen Bedingungen ist, und auch dass frühere Versuche zur Modellierung des Phänomens mit molekulardynamischen Simulationen gescheitert sind, motiviert die Entwicklung eines dem Superspreading gewidmeten Kraftfeldes. Anwendungen in groß-skaligen Simulationen von Tropfenausbreitung weisen einen glatten Übergang an der Kontaklinie bei Superspreadingbedingungen auf. Es wird gezeigt, dass diese Beobachtung plausible Erklärungen für experimentelle Ergebnisse und eine stimmige Beschreibung des Superspreadingmechanismus liefert.

Während die dynamischen Eigenschaften und Mechanismen des Massetransports von molekularen Präkursoren verstanden sind, sind die Bedingungen, die zur Bildung von Präkursoren oder verschiedenartig geformter Präkursoren führen, umstritten. Groß-skalige Simulationen von Tropfenausbreitung, neue Analysemethoden, und umfassende Freie Energie Berechnungen werfen Licht auf diese Fragestellungen und lösen den Konflikt über die Rolle des Spreitparameters bei der Bildung von Präkursoren.

Acknowledgments

My special thanks goes to my thesis adviser Prof. Ahmed Ismail. The work at hand was decisively influenced by the freedom that he granted to choose research topics and to develop ideas. His continuous encouragement, investments, and patience were of great support. His serenity and optimism were important backup and motivation especially after failures.

I thank Prof. André Bardow for taking the role as the senior supervisor that is required by the AICES program and for acting as chairperson in my doctoral committee. Although our meetings were not rich in number, they were rich in quality and crucial impulses were given here. For his willingness to act as the reviewer for this thesis I acknowledge Prof. Martin Horsch.

Steve Plimpton, Axel Kohlmeyer, Paul Crozier, and Stan Moore are acknowledged for helpful communication concerning all aspects related to LAMMPS. Moreover, they deserve gratitude for their continuous effort to develop and maintain this code. All MD simulations reported here were performed with this software, and the thesis could not exist in its present state without people like them. Likewise, this acknowledgment is directed to them substitutional for everybody who provides similar service to the scientific computing community.

Jonathan Halverson, whose work on superspreading was one of the most important starting points of my own work, is acknowledged for helpful discussions. I thank Francisco Fontenele for broadening my horizon by calling my attention to the physics literature. Jeff Hammond is acknowledged for providing initial access to the high-performance computers at the ALCF. I thank Benjamin Berkels for the idea and the support in the application of the moment-based surface analysis. Likewise, Brooks Rabideau and Tod Pascal deserve gratitude for their support with the application of the 2PT method. Markus Schmidt is acknowledged for experimental support.

Research needs a solid infrastructure. For providing this infrastructure, I thank the entire AICES service team, especially Dr. Nicole Faber, Joëlle Janssen, Miriam Hagelstein, and Nadine Bachem, and the entire AICES IT team, especially Christoph Stolz, Christian Delzepich, and Philip Knott.

On top of infrastructure, scientific work is dependent on resources and funding agencies. Accordingly, I acknowledge the DFG for the funding for me, and JARA-HPC, PRACE, LRZ, and ALCF for granting access to computer resources at ITC at RWTH Aachen, Hydra and SuperMUC in Garching, and Vesta at ANL. Likewise, I thank the local IT teams of these locations as well as the JSC in Jülich for support with the usage of these machines.

The work reported here was supported by students that I have supervised. In particular, the work on real-space error estimates and tabulation of pair potentials was done by Wayne Mitchell, the interlaced algorithm was implemented by Thomas Ziob, and the comparison of the interlaced and noninterlaced PPPM method in practice was done by Alexander Jacobsen. I thank them, as well as Fabian Key, Tina Chen, and Zhijin Wu, for the positive experience of having worked with them.

I thank my office mates Jochen Schütz, Christian Waluga, Aravind Balan, Mohammad Zakerzadeh, Farshad Roohbakhshan, Michael Woopen, and especially Kareem Noureldin and Janine Mergel, as well as my group mates Marcus Schmidt, Carl-Simon Adorf, Amir Niazi, Paul Springer, Daniel Tameling, Diego Fabregat-Traver, Martin Carballo Pacheco, Brooks Rabideau, and Pan Chen for establishing enjoyable and motivating working conditions.

Finally, even though I feel that this is not the right place to for private matter, I want to express that I am particularly grateful to my parents Marie-Luise and Ferdinand, my partner Anne, and my son Raphael.

Contents

1 Introduction **1**

1.1 Long-ranged interactions in molecular simulation and interface science . 2

1.2 Thesis objectives and overview 4

2 The PPPM for dispersion in theory and practice **6**

2.1 Long-range dispersion solvers for molecular simulation 6

2.2 Ewald summation for dispersion 9

2.3 The structure factor and mixing rules 14

2.4 The particle-particle particle-mesh algorithm 16

2.4.1 Discretization of dispersion coefficients and transformation to reciprocal space 19

2.4.2 Differentiation and backinterpolation 20

2.4.3 The optimized Green's function $\hat{G}_{\text{opt}}$ 22

2.4.4 Interlacing 24

2.4.5 Error estimates 25

2.5 Numerical tests: accuracy and single-core performance 27

2.5.1 Accuracy of variants of the PPPM method 28

2.5.2 Accuracy of error estimates 31

2.5.3 Performance comparison of variants of the PPPM method . . . 31

2.5.4 Scaling behavior compared to Ewald summation 35

2.6 Impact of the real- and reciprocal-space errors on physical behavior . . 36

2.6.1 Parameter study with hexane 36

2.6.2 Analysis of the errors 39

2.7 Influence of the PPPM method on physical properties 44

2.7.1 Lennard-Jones particles 44

2.7.2 SPC/E water 45

2.7.3 Hexane 48

2.8 Increased performance by using long-range dispersion solvers 49

2.8.1 Performance tuning with flexible choice of cutoffs 50

2.8.2 Performance simulations 56

2.8.3 Discussion 61
2.9 Interlacing vs. non-interlacing for dispersion in practice 63
2.10 Summary 66

3 The molecular mechanism of superspreading 68
3.1 Literature review and motivation 68
3.2 Molecular modeling of surfactant mixtures 72
3.2.1 Modeling depth and initial potentials 72
3.2.2 Functional form 74
3.2.3 Fitting procedure 75
3.2.4 Force field validation with model molecules 79
3.2.5 Discussion of the fitting strategies 88
3.2.6 Model validation with surfactant simulations 89
3.2.7 Assessment of the developed force field 95
3.3 Simulations of surfactant enhanced spreading 96
3.3.1 Simulation setup and analysis methods 96
3.3.2 Wetting dynamics and droplet shape 104
3.3.3 Relevance for superspreading 113
3.4 Summary and concluding remarks 116

4 Molecular precursors 119
4.1 Previous studies and open questions 119
4.1.1 Experimental observations 119
4.1.2 Theoretical models 120
4.1.3 Findings from molecular dynamics simulations 122
4.1.4 Objectives 124
4.2 Simulations of molecular precursors 126
4.2.1 Simulation setup 126
4.2.2 Analysis methods 131
4.2.3 Precursor types, transition regimes, and wetting dynamics . . . 135
4.2.4 Effects of correct and incorrect precursor analysis 146
4.3 Requirements for precursor formation 148
4.3.1 The spreading coefficient 148
4.3.2 Free energy of depositing fluid on substrates 150
4.4 Summary 156

5 Summary and future research 158

A Surface tensions and liquid densities from slab simulations 162

B Force field parameters 164

B.1 Nonbonded interactions 164

B.2 Bonded interactions 168

Bibliography 172

List of Figures

1.1 A droplet on a solid substrate 2

2.1 Principal idea of the Ewald method 10
2.2 Physical interpretation of the Ewald sum 11
2.3 Accuracy of the different versions of PPPM for dispersion 29
2.4 Accuracy of PPPM for random particle positions and equilibrated systems 30
2.5 Comparison of the accuracy obtained with normal and interlaced PPPM 30
2.6 Comparison of the real-space error estimate with measured force errors 32
2.7 Comparison of the reciprocal-space error estimate with measured force errors 33
2.8 Reciprocal-space error estimate in interfacial systems 33
2.9 Single-core timing of PPPM variants at varying force accuracy 34
2.10 Scaling of Ewald summation and PPPM on a single-core 35
2.11 Influence of the PPPM parameters on simulated surface tensions and liquid densities I 38
2.12 Influence of the PPPM parameters on simulate surface tensions and liquid densities II 39
2.13 Hexane surface and abstraction that highlights the importance of the real-space error 41
2.14 Measured local force errors in an interfacial system 43
2.15 Density profiles of an interfacial LJ system 46
2.16 Surface tensions of hexane over the cutoff 49
2.17 Adjustment of the real-space error with the Ewald parameter and the cutoff 51
2.18 Simulated liquid densities when using small cutoffs 53
2.19 Splitting of pair potentials with and without rRESPA 54
2.20 PPPM performance on a BG/Q architecture 59
2.21 PPPM performance on the Sandy Bridge architecture at SuperMUC . . 60

3.1 Alkyl ethoxylate, trisiloxane, and perfluoroalkane surfactants 77
3.2 Conformers of the trisiloxane surfactant 78

3.3 Excess quantities of mixtures of perfluorohexane and hexane 83
3.4 Experimental and simulated densities of DME–water mixtures 85
3.5 Radial distribution functions of DME–water mixtures 86
3.6 Viscosities and diffusion coefficients of DME–water mixtures 87
3.7 Histograms of the z position of central surfactant atoms 91
3.8 Local densities in simulations of surfactants at the water surface 92
3.9 Surface tensions of the surfactant-laden water surface 94
3.10 Starting configuration of simulations of surfactant assisted wetting . . . 99
3.11 Analysis steps of simulations of surfactant assisted spreading 100
3.12 Classification of the local droplet shape 102
3.13 Example of moment-based surface analysis 104
3.14 Base radii r in simulations of surfactant enhanced spreading 106
3.15 Snapshots from different spreading regimes 107
3.16 Surfactant density at the solid–liquid interface in surfactant assisted spreading 109
3.17 Surfactant density close to the three-phase contact line 109
3.18 Droplet snapshots in the viscous regime 110
3.19 Droplet shapes in the viscous regime color-coded with the classifier $C(x)$ 111
3.20 Maximum droplet classifier $C_{\max}$ over the entire viscous spreading regime 112
3.21 Proposed mechanisms for superspreading and usual surfactant enhanced wetting 114
3.22 Snapshot of the contact line region for the superspreading and non-superspreading simulations 115

4.1 Sketches of different precursor types 120
4.2 Definition of the precursor length 125
4.3 LJ and modified Buckingham potential 129
4.4 Equilibrated slab system of the atomistic fluid 129
4.5 Spreading process of chain molecules 130
4.6 Number counts of liquid particles over the z direction 131
4.7 Definition of the precursor length 132
4.8 Sketch of a continuously growing precursor and a single layer precursor with markers for the length of the layers l_1 and l_2 133
4.9 Measurement of the droplet profile to mimic ellipsometric experiments . 134
4.10 Evolution of the fist four layers of the chain molecules 136
4.11 Effective thickness of the droplet of chain molecules 137
4.12 Evolution of the first layers of r_i of the chain molecules 138

4.13 Evolution of the first four layers of the atomistic fluid 140
4.14 Effective thickness of droplets of the atomistic fluid 141
4.15 Detection of the crossover point for the atomistic fluid at low substrate energies 142
4.16 Measured values of r_1 and fitted functions 143
4.17 Spreading exponent obtained from the fit to r_1 for the chain molecules 144
4.18 Analysis of the spreading exponent from l_1 145
4.19 Simulation setup for the computation of the spreading coefficient 149
4.20 Spreading coefficients as a function of the substrate energy 150
4.21 Solid substrates covered with thin layers of liquid 152
4.22 Change in free energy of depositing layers of the atomistic fluid on a bare substrate 153
4.23 Change in free energy of depositing layers of the chain molecule on a bare substrate 154
4.24 Changes in free energy of depositing layers of the chain molecule on a bare substrate with corrected free energy of the bulk 155
4.25 Schematic explanation of the role of the spreading coefficient 156

B.1 Surfactants with labels to identify force field parameters 165

List of Tables

2.1 Comparison of simple truncation and PPPM for a LJ system 45

2.2 Comparison of simple truncation and PPPM for water 47

2.3 Comparison of simple truncation and PPPM for hexane 48

2.4 PPPM parameters in simulations to find the lower bound for the cutoff 52

2.5 PPPM settings in simulations to validate methods to improve computational efficiency 55

2.6 Surface tensions and liquid densities of hexane obtained with acceleration settings 56

2.7 Surface tensions and liquid densities of water obtained with acceleration settings 56

2.8 Comparison of the interlaced and non-interlaced algorithm in practical applications 65

3.1 Experimental and simulated quantities of model compounds 81

3.2 Interfacial tensions of model compounds and water 88

3.3 Surface area per surfactant molecule at the interfaces at maximum packing 95

3.4 Setup for surfactant enhanced spreading simulations 98

3.5 Spreading exponents and duration of different spreading regimes 105

B.1 Partial charges and dispersion coefficients 166

B.2 vdW coefficients for the repulsive terms 167

B.3 Parameters for bond potentials 169

B.4 Parameters for angle potentials 170

B.5 Parameters for dihedral potentials 171

1 Introduction

As vividly demonstrated in the preface of "Capillarity and Wetting Phenomena" [1], wetting and spreading phenomena are ubiquitous in daily life and nature, but also in numerous industrial applications, such as lubrication [2], herbicides and pesticides [3], printing [4], oil recovery [5], self-cleaning surfaces [6], and coatings [7]. Because of its importance in various disciplines, wetting of liquids on solids has been an intense field of study ever since the groundbreaking work of Young more than 200 years ago [8].

From a macroscopic, equilibrium point of view, the wetting behavior of droplets can be well characterized as droplets having spherical caps whose contact angles θ can be described by Young's equation[1]

$$0 = \gamma_{sl} - \gamma_s + \gamma_l \cos\theta, \tag{1.1}$$

which can be understood as a force balance of the solid–gas, solid–liquid, and liquid–gas surface free energies or surface tensions γ_s, γ_{sl}, and γ_l as depicted in Figure 1.1. If $\gamma_{sl} - \gamma_s < \gamma_l$, the droplet is in the partial, incomplete, or non-wetting regime and Young's equation predicts a finite value for the contact angle θ. Otherwise, the droplet completely wets out to a thin film making neither the definition of θ sensible nor the computation with Young's equation possible. This regime is the so-called (complete) wetting regime. The different regimes can easily be distinguished by their spreading coefficient,

$$S = \gamma_s - \gamma_{sl} - \gamma_l, \tag{1.2}$$

that changes sign from negative to positive upon transition from partial to complete wetting.

In contrast to the macroscopic equilibrium picture, even droplets of the simplest liquids can show a wealth of phenomena when considering the microscopic scale or non-equilibrium situations, i.e. wetting or dewetting processes [10, 11], which renders a proper description of all details of spreading challenging with purely macroscopic approaches. In fact, even the seemingly simple case of the spreading of a non-volatile liquid on a macroscopically smooth substrate strains continuum theories to their limits:

[1] Despite the controversial discussion on the terminology [9], the term Young's equation is used for Equation (1.1) because it is the commonly used expression.

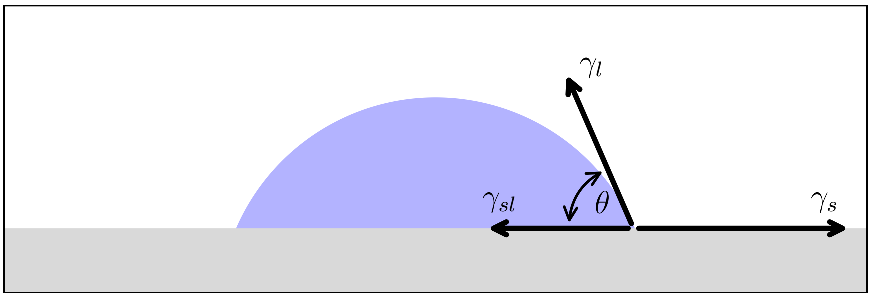


Figure 1.1: A droplet on a solid substrate. The contact angle θ is determined from a force balance of the solid–liquid, liquid, and solid surface energies γ_{sl}, γ_l, and γ_s.

Applying no-slip boundary condition and lubrication theory to the spreading of droplets with a sharp leading edge predicts logarithmically diverging viscous dissipation at the contact line. In other words, continuum theory predicts that droplets cannot spread on solid substrates, which disagrees with everyday observations and is therefore known as the Huh-Scriven paradox [10, 12]. A loophole to this quandary is to impose other boundary conditions, such as slip conditions, and in this way obtain stable numerical behavior.

Although imposing boundary conditions motivated not by physical requirements but by numerics is not necessarily a good idea, as discussed in [13], this procedure has been used successfully to describe many wetting phenomena that have microscopic aspects with macroscopic approaches and continuum theories. There are, however, wetting phenomena in which the continuum view is indefensible and molecular details have to be considered for a proper description and understanding. Because virtually all studies on wetting phenomena have been performed using experimental techniques or modeling from the macroscopic point of view, insight into these phenomena is very limited, because the relevant length scales cannot be resolved appropriately. The study at hand focuses on these phenomena with clear molecular character. The used technique is the molecular dynamics (MD) simulation method, which resolves the molecular scales and provides information inaccessible to experimental approaches.

1.1 Long-ranged interactions in molecular simulation and interface science

Because MD simulations are at the heart of this work, this section provides a brief description of the foundations of this simulation method. It will also be outlined how the viewpoint on long-ranged interactions in the molecular simulation community conflicts with the viewpoint from the interfacial science community, which will be the motivation for a major part of the work described in this study.

Classical MD simulates the trajectories of atoms or coarse-grained beads. Trajectories are generated by computing the interactions between the atoms and forward integrating their positions over time using Newtonian mechanics or modifications, for example to sample desired temperatures or pressures [14].

The physically correct, but often unaffordable way to describe molecular interactions in an ensemble of atoms is to solve Schrödinger's equation. Instead, interactions between molecules are typically modeled with potentials that contain several contributions, such as bonded potentials for nearest neighbors in molecules, and nonbonded potentials for interactions between atoms of different molecules or atoms within the same molecule separated by multiple bonds. This approach still leaves a lot of freedom for modeling, including the functional forms used for the interactions and whether polarization or multi-body effects should be considered.

Here we examine a simple, but successful and therefore also common approach, in which nonbonded interactions are modeled as pairwise potentials by a combination of Coulomb interactions to model electrostatics,

$$u_{\text{coul}} = -\frac{1}{4\pi\epsilon_0}\frac{q_i q_j}{r}, \tag{1.3}$$

where ϵ_0 is the vacuum permittivity, q_i and q_j are the partial charges of the atoms i and j, and r is their distance, a dispersion term

$$u_{\text{disp}} = -\frac{C_{ij}}{r^6}, \tag{1.4}$$

where C_{ij} is the dispersion coefficient that describes the strength of the interactions, and a repulsive term u_{rep} that models the interactions related to the Pauli principle and decays very quickly. The success of this approach stems from that although many physical effects are neglected, the three most important contributions are included. Moreover, both the usage of the r^{-1} and the r^{-6} functionals for the Coulomb and the dispersion interactions have a strong physical foundation [15].

The number of pairs of atoms scales quadratically with the number of particles in a system, which makes evaluation of all of these pair potentials computationally infeasible even for moderately large systems. The issue is overcome by a treatment of the potential according to whether it is short- or long-ranged. From a mathematical point of view, a pair potential u_p is considered short-ranged if the sum of the potential interactions of an atom with all other atoms in an infinite system is absolutely convergent; otherwise, the potential is considered long-ranged. According to this definition, electrostatics are long-ranged and dispersion interactions short-ranged. As a consequence, the common approach in MD nowadays is to simply truncate dispersion interactions at a cutoff distance and using long-range solvers to treat electrostatics [14].

The classification of long-ranged and short-ranged forces changes dramatically when changing the point of view: Because electrostatics involve attractive and repulsive interactions, shielding effects and formation of double layers can lower the effective interactions. As a consequence, the interactions between charge neutral mesoscopic objects caused by electrostatic interactions decay exponentially. In contrast to that, dispersion interactions are always attractive and therefore add up. The interactions between surfaces caused by dispersion interactions can therefore scale as strongly as $1/D$, where D is the distance between the surfaces, depending on the surface geometry and therefore can have effects on the range of tens of nanometers [15]. In interfacial science, in particular in the context of wetting, the classification from the molecular simulation community is therefore inverted: electrostatics are considered short-ranged, whereas dispersion is considered long-ranged.

1.2 Thesis objectives and overview

That dispersion interactions in interfacial systems are long-ranged, at least over a distance that is much larger than typically used cutoff values, makes their incorporation in MD simulations essential for accurate modeling. Because MD has historically mainly been developed for systems without interfaces, methods for this purpose are underdeveloped compared to the treatment of electrostatics. In Chapter 2 it will be shown that the currently available tools for incorporating long-range dispersion interactions are not sufficient for the targeted wetting studies and an improved algorithm for their incorporation, the PPPM method for dispersion, will be presented. Both theoretical features of the algorithm as well as information relevant to its application in practice will be described. Because the applicability of this algorithm is not restricted to molecular wetting phenomena but can be beneficial in all kinds of molecular simulations, the algorithm is presented in greater detail than strictly necessary to understand the subsequent chapters on wetting phenomena.

The developed algorithm is then applied to study selected wetting phenomena with clear molecular aspects, which is the primary objective of this study. In particular, the phenomenon of superspreading, the ultra-rapid wetting of aqueous solutions facilitated by trisiloxane surfactants [3], and the formation of molecular precursors, the development of films of molecular thickness that can precede spreading droplets [16], are studied.

The phenomenon of superspreading is not only characterized by contact angles that are too small to be measured in experiment, but also by the violation of “Tanner’s law” [17], a power law that relates the spreading radius to the spreading time that is almost universally valid for spreading droplets. There is a general consensus that this process is facilitated by a molecular mechanism of the involved surfactants. Because the relevant length scales at which this mechanism occurs cannot be observed in experiment, the nature of this mechanism is still under debate [18]. In contrast to experiment, the relevant length scales are resolved in MD simulations and molecular mechanisms can be observed directly from the simulation output. Work that aims to capture this mechanism in MD and to identify it is presented in Chapter 3.

The phenomenon of molecular precursors is not fully understood despite the continuous effort in the 25 years since the discovery of the phenomenon [19,20]. In particular, theoretical models that can capture all features known from experiment are not yet available. Moreover, the fundamental questions of under which conditions precursors form and what drives the development of precursors of different shapes is not even understood rudimentary. The objective of the work on precursors is to identify the requirements for the formation of molecular precursors. As a byproduct, conflicting findings from the literature and several misleading statements caused by inaccurate analysis of previous experiments and simulations are identified. This work is presented in Chapter 4

The highlights of the work presented in Chapters 2 to 4 are briefly summarized in Chapter 5. Further areas of research are proposed based on these findings.

2 The PPPM for dispersion in theory and practice

2.1 Long-range dispersion solvers for molecular simulation

Although dispersion interactions are formally short-ranged according to the mathematical definition, they can still have effects on the scale of tens of nanometers [15]. For reasons briefly discussed in Section 1.1, however, dispersion interactions have typically been truncated for distances beyond a cutoff of around 1 to 2 nm in MD simulations. For homogeneous systems, the error introduced by this truncation is usually acceptable because dispersion interactions decay quickly, errors in neglected forces partially cancel, and errors in energies and pressures can be corrected efficiently and accurately during the simulation [14]. For inhomogeneous systems, however, especially those involving interfaces, the conditions that justify the simple truncation of dispersion interactions are no longer met, which has led to the development of various correction methods that are briefly described in the following.

The need for inclusion of long-range dispersion forces in molecular simulations in interfacial systems was already recognized by Chapela et al. [21] more than 35 years ago and has motivated the *a posteriori* correction method developed by them and later corrected by Blokhuis et al. [22]. Specifically, they performed simulations at the liquid–vapor interface and introduced a correction term for the surface tension in which missing contributions of the truncated part of the dispersion interactions are added based on the density distribution of the particles. The disadvantage of this method is that it does not correct the dynamics of a simulated system, but rather only provides corrections for observables. Because simulated density profiles are sensitive to truncated dispersion forces, the accuracy of this correction method is limited by the accuracy with which the density profile can be computed. Moreover, this method averages out instantaneous fluctuations at the interface that can arise because of capillary waves. As a consequence, the results obtained with this correction method are different from what would be obtained with full incorporation of long-range dispersion forces in the limit of very large cutoffs or more elaborate long-range solvers [23].

The logical continuation of this method, an on-line correction based on density profiles was introduced almost 30 years later by Janeček [24] and extended by Werth et al. [25] to multi-site Lennard-Jones (LJ) models. This method estimates truncated dispersion forces based on the current density. Compared to the *a posteriori* correction method, it was shown that these correction methods provide accurate results for relatively simple liquids and can result in fast computations for planar interfaces [24, 25]. The major drawback of this method is that while an on-line correction based on the density distribution can in principle be applied to arbitrarily shaped systems, efficient implementations require an *a priori* knowledge about the shape of the interface and have hitherto only been developed for planar interfaces. Moreover, the accuracy of this method for complex mixtures, systems with surfactants, or systems close to the critical point, where stronger fluctuations at the interface are expected, is questionable.

An alternative to the correction methods based on local densities are methods based on modifications of the cutoff in simulations of interfacial or inhomogeneous systems. The conceptually simplest idea is to use a sufficiently large cutoff such that truncation effects become negligible. Zubillaga et al. [26] showed that when the cutoff is increased above a certain threshold, errors because of truncating the dispersion interactions have a negligible influence on simulated quantities. In their study with the atomistic OPLS-AA force field [27] this threshold was at approximately 23 Å. Using extremely large cutoffs is simple and applicable with almost no restrictions, ensures highly accurate results, and requires little overhead in terms of code development. However, as the computing time needed to evaluate all pair interactions scales with $\mathcal{O}(r_c^3)$, where r_c is the cutoff, this approach is extremely unattractive computationally. Other methods based on modifications of the cutoff are based on twin-ranged cutoffs [28] in which simulations are executed with a small and a large cutoff. Interactions between particles farther apart than the inner cutoff but closer than the outer cutoff are computed less frequently. A final approach based on cutoffs is the usage of adaptive cutoffs proposed by Wang et al. [29], which uses larger cutoffs in regions of strong inhomogeneity. As shown by Springer [30], this method provides highly scalable computations. The fully automated detection of regions where increased cutoffs are required, however, is difficult and requires further development for applicability of this method to arbitrary systems.

The third class of methods are based on the separation or expansion of pair potentials in multiple terms. Members of this third class of method are the fast mutlipole method (FMM) [31], the multilevel summation method (MSM) [32], Ewald summation [33], and grid-based methods that are based on Ewald summation, namely the particle-particle particle-mesh (PPPM) [34] and particle-mesh-Ewald (PME) methods [35, 36].

In contrast to the previously described methods, each of these was originally developed for Coulomb interactions. A common feature of these methods is that one of the splitted terms is computed in the traditional way over a pairwise sum of closely spaced particles, while the other terms are treated with approaches that are different for each of these methods. The gain in accuracy results from that the part computed via direct pairwise summation within a given cutoff decays quickly to zero such that the truncation part becomes negligible with an appropriate choice of parameters, or even identically zero as for the MSM. A strong upside of the FMM and the MSM is that the scaling of the computational costs is $\mathcal{O}(N)$ [37], where N is the number of particles, and both methods require mainly local communication. The MSM has recently been extended to dispersion interactions by Tameling et al. [38]. An extension of the FMM to dispersion interactions is presented in [39]. The extension of Ewald summation to dispersion interactions was developed for dispersion by Williams [40], Perram [41], and Karasawa and Goddard [42], and later applied to surface simulations for example by Mecke et al. [43], López-Lemus et al. [44], in't Veld et al. [23, 45], Ou-Yang et al. [46], and Alejandre and Chapela [47].

The wetting phenomena studied in this thesis include spreading simulations in which no information on the shape of the droplets is available beforehand. Given the discussion above, the application of the methods based on corrections of the density is therefore not promising to obtain accurate and fast computations, while using large cutoffs or twin range-cutoffs will impose an unacceptable performance burden. The dynamic cutoff method might be promising, yet, it is still not sufficiently developed to allow application to arbitrary systems. Moreover, in some systems, e.g., systems with a molecular precursor, the interfacial area will be large. The spatial region in which large cutoffs will have to be employed will then be a substantial fraction of the total domain which causes a considerable performance penalty. Consequently, one of the more elaborate methods that were originally developed for Coulomb interactions will be required. Because among these algorithms the PPPM method provides fast computations in many applications [37], is more easy to implement than the other methods, and is also very widespread for Coulomb interactions, this is the method of choice applied for long-range dispersion interactions in this study.

Grid-based Ewald methods have already been applied to dispersion interactions by Essmann et al. [36] and Wennberg et al. [48] for the PME, and by Shi et al. [49] for PPPM. The study of Essmann et al. focused mainly on Coulomb interactions; the dispersion part played a subordinate role in their work. The study of Wennberg et al. was mainly application-oriented. The study of Shi et al. places a stronger focus on PPPM and dispersion interactions. The description of the algorithm, however, is incomplete.

For example, the equations for the virial in their work is complicated and involves unnecessary computations for which simpler equations are provided here. Moreover, they implemented and used the algorithm, but did not study how this algorithm should be applied in practice. As a consequence of this, their simulation were technically flawed, as shown below, and disagreed with other work.

The remainder of this chapter is divided into two parts. The first part will provide a theoretical description of the PPPM method for dispersion. A brief description of the Ewald method for dispersion, which is the approach that is underlying the PPPM algorithm, will be given in Section 2.2. After discussing the role of the mixing rule in application of Ewald solvers in Section 2.3, the PPPM algorithm will be presented in Section 2.4 and numerical tests that describe its theoretical properties are presented in Section 2.5.

The second part will provide information on how the PPPM method for dispersion should be used in practice and what its practical benefits are. Section 2.6 describes how the two main errors of the method influence physical results. It will also be pointed out that their influence is very different compared to PPPM for Coulomb interactions. It will be shown that this is crucial in application of the solver to obtain efficient and accurate simulations. Afterwards, we address how important the inclusion of long-range dispersion interactions is to get an accurate description of different systems in Section 2.7. In Section 2.8 it will be demonstrated that in contrast to the common belief that PPPM for dispersion imposes a performance penalty, simulations with this algorithm can be faster that those using a simple truncation. Finally, we briefly compare two variants of PPPM in Section 2.9 in practical applications before providing a summary of the main findings in Section 2.10.

2.2 Ewald summation for dispersion

From a mathematical point of view, the principal idea behind Ewald methods (cf. Figure 2.1) is to split up a potential

$$u_p(r) = \frac{1}{r^p} = \frac{f_p(\beta_p, r)}{r^p} + \frac{1 - f_p(\beta_p, r)}{r^p}, \tag{2.1}$$

where p is a positive integer, β_p is the Ewald coefficient, and $f(\beta_p, r)$ is a splitting function. The first term on the right-hand side of Equation (2.1), which is also referred to as the real-space term, is solved by direct evaluation of the pair potential inside a chosen cutoff radius. The second term is called the reciprocal-space or Fourier term and is solved using a Fourier transform.

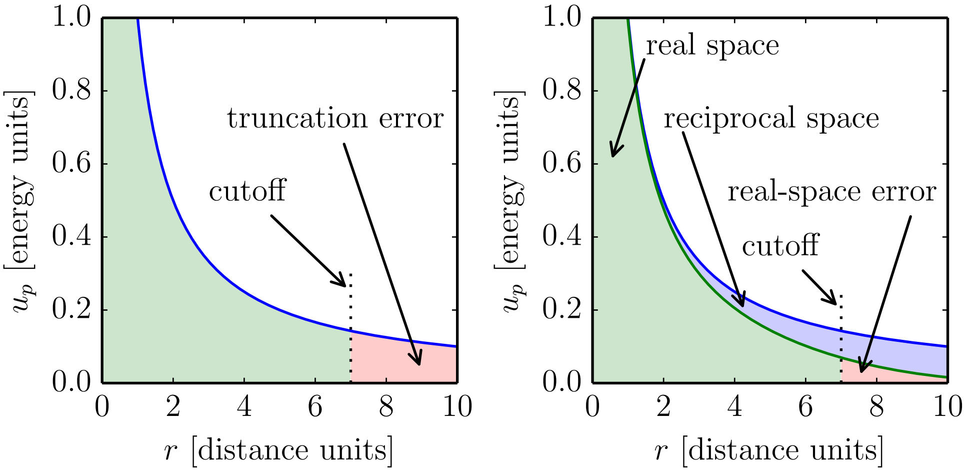


Figure 2.1: Schematic comparison of truncation and Ewald treatment of pair potentials. Left: truncation of a pair potential u_p at the cutoff. The red area under the curve beyond the cutoff is related to the error in the computations that arises from the truncation. Right: treatment with the Ewald method. The potential is split into two terms. The real-space error caused by truncating the real-space term is strongly decreased versus pure truncation of the potential. For completeness it is mentioned that there is also a reciprocal-space error that cannot be visualized easily and is therefore not depicted.

The splitting function $f_p(\beta_p, r)$ is in principle arbitrary. As outlined in [50], however, desirable qualities are: (i) that the function decays sufficiently rapidly for increasing r such that the contribution of the real-space term beyond a cutoff can be neglected and (ii) that $(1 - f_p(\beta_p, r))/r^p$ is a slowly varying function of r, such that the Fourier transform converges for a small number of included reciprocal-space vectors. These requirements in principle leave an infinite number of reasonable choices for the splitting function, of which a few are discussed by Heyes [51]. The most common choice is to derive the splitting function from the Euler gamma function [36, 40, 52],

$$f_p(\beta_p, r) = \frac{1}{\Gamma(p/2)} \int_{(r\beta_p)^2}^{\infty} t^{p/2-1} e^{-t} dt, \tag{2.2}$$

where $\Gamma(p/2)$ is the Euler gamma function

$$\Gamma(p/2) = \int_0^{\infty} t^{p/2-1} e^{-t} dt. \tag{2.3}$$

This choice of the splitting function fulfills the requirements above. Moreover, it has the advantage that the mathematics works well for any value of p.

In addition to the mathematical interpretation given above, there is a physical interpretation of the Ewald summation [14]: as depicted in Figure 2.2, the physical interpretation is to superposition a set of given discrete interaction sites with a set of counterbalancing smeared sites. The real-space term then contains the discrete interaction sites and the counterbalancing smeared sites that shield the interactions of the discrete sites and therefore render them much more short-ranged. The reciprocal-space term contains minus the smeared interaction sites whose contributions have been added to the real-space term. The exact form of the splitting function and the resulting equations is then a product of the choice of the function that is used to smear the discrete interaction sites [51]. For the case of Coulomb interactions with $p = 1$, the typically used ansatz for the splitting function described above corresponds to smear the charges with a Gaussian bell curve [50].

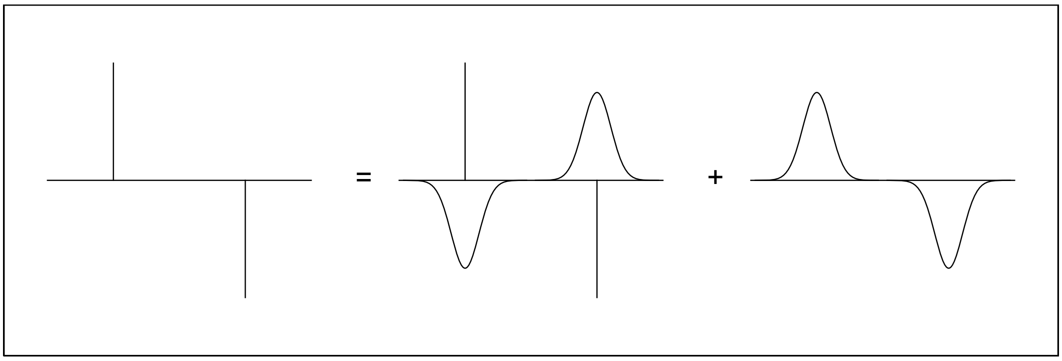


Figure 2.2: Physical interpretation of Ewald summation: A set of discrete interaction sites is superpositioned with a set of smeared interaction sites.

The hitherto more general discussion on the Ewald sum will in the following be strongly narrowed down: because of the utilization of the Fourier transform, Ewald summation is restricted to systems with periodic boundary conditions, which is why a three-dimensional periodic box will be assumed in the following. Moreover, the box is assumed to be rectangular with volume V and lengths $\mathbf{L} = (L_x, L_y, L_z)$, where x, y, and z specify the Cartesian coordinates. In addition, the pairwise r^{-6} dispersion potential given in Section 1.1 is assumed. Finally, the conventional choice of the splitting function derived from the Euler gamma function described above is made. Because the method is only presented for dispersion interactions with $p = 6$, the p indices will be dropped in the following when it is clear that dispersion is meant from the context. Moreover, we restrict the general description of the method to the case where the dispersion coefficients C_{ij} in Equation (1.4) follow a geometric mixing rule:

$$C_{ij} = \sqrt{C_{ii}C_{jj}} = c_i c_j, \tag{2.4}$$

where c_i and c_j are atom-specific constants. A generalization will be provided in Section 2.3.

Taking these assumptions, the dispersion energy E of the particles within the unit cell can be obtained by taking a sum over the pair interactions of all atoms in the cell with all other atoms in the same cell and periodic neighbors to obtain

$$E = \frac{1}{2} \sum_{i=1}^{N} \sum_{j'=1}^{N} \sum_{\mathbf{n} \in \mathbb{Z}^3} \frac{C_{ij}}{r_{ijn}^6}, \tag{2.5}$$

with

$$r_{ijn} = \|\mathbf{r}_i - \mathbf{r}_j + \mathbf{n} \circ \mathbf{L}\|, \tag{2.6}$$

where the first two sums are over all particles in the system and the third sum is over all periodic boxes, and $\mathbf{r}_i$ and $\mathbf{r}_j$ are the coordinates of particles i and j. The prime in the middle sum indicates that the self-interaction $i = j$ for $\mathbf{n} = \mathbf{0}$ is omitted. The $\circ$ indicates element-wise multiplication; the result of the multiplication is thus a vector of size 3. When expanding Equation (2.5) with the splitting function above and transforming the second part of the sum to Fourier space as described in detail in [36], one obtains for the dispersion energy of the system

$$E = E^{(r)} + E^{(k)} + E^{(0)} + E^{(s)}, \tag{2.7}$$

with

$$\begin{aligned}
E^{(r)} &= \frac{\beta^6}{2} \sum_{i=1}^{N} \sum_{j'=1}^{N} \sum_{\mathbf{n} \in \mathbb{Z}^3} C_{ij} \left(1 + \beta^2 r_{ijn}^2 + \frac{\beta^4 r_{ijn}^4}{2}\right) \frac{\exp\left(-r_{ijn}^2 \beta^2\right)}{\beta^6 r_{ijn}^6}, && (2.8)\\
E^{(k)} &= \frac{1}{2V} \sum_{\mathbf{h} \neq 0} \tilde{G}(\mathbf{h}) \tilde{\mathcal{S}}^2(\mathbf{h}), && (2.9)\\
E^{(0)} &= \frac{\pi^{3/2} \beta^3}{6V} \sum_{i=1}^{N} \sum_{j=1}^{N} C_{ij}, && (2.10)\\
E^{(s)} &= -\frac{\beta^6}{12} \sum_{i=1}^{N} C_{ii}, && (2.11)
\end{aligned}$$

where

$$\tilde{G}(\mathbf{h}) = \frac{\pi^{2/3}}{12} h^3 \left(\pi^{1/2} \mathrm{erfc}(b) + \left(\frac{1}{2b^3} - \frac{1}{b}\right) e^{-b^2}\right) \tag{2.12}$$

is the influence function or Green's function, the $\mathbf{h}$-vectors are the reciprocal space vectors that are in the set $\mathbf{h} \in \{2\pi\mathbf{n} \oslash \mathbf{L} : \mathbf{n} \in \mathbb{Z}^3\}$ and $h = \|\mathbf{h}\|$ and $b = h/(2\beta)$. The

$\oslash$ indicates element wise division.

The first term $E^{(r)}$ in Equation (2.7) is the real-space term. Its contribution is solved in real space with the conventional approach of a sum over pairs of particles. As for usual pair potentials, the sum in Equation (2.8) is infinite and would have to be computed over all pairs of particles to get a mathematically exact result. Because this is prohibitively expensive, the conventional method of only considering contributions of particles that are closer than a chosen cutoff is applied. The upside compared to the original, non-split potential is that the real-space potential decays much more rapidly with increasing particle distance. The error introduced by truncating the potential is therefore much more tolerable — and, for a good choice of the Ewald parameter and the cutoff, even negligible, as will be shown later.

The second term $E^{(k)}$ is the Fourier space term. $\tilde{\mathcal{S}}$ is a function that is dependent on the position of the particles and their interaction energies and is computed using the Fourier transform. As with Equation (2.8), the sum in this equation, which is over all reciprocal-space vectors $\mathbf{h}$ except $\mathbf{h} = 0$, is infinite and has to be truncated in application. To obtain accurate results, the truncation of the reciprocal-space vectors has to be performed such that a sufficient number of reciprocal-space vectors is maintained and that the truncation of the vectors is performed along the surface of a sphere. Further details on $\tilde{\mathcal{S}}$ in Equation (2.9) will be provided in Section 2.3.

The third term $E^{(0)}$ contains the contributions that arise from the contribution of the reciprocal space term for $\mathbf{h} = 0$. In contrast to the Ewald sum for Coulomb interactions, for which this term is dependent on the boundary conditions that are assumed outside the truncated part of the reciprocal-space vectors, this term does not depend on the environment for the dispersion case. The fourth term $E^{(s)}$ corrects for the self-energy that arises from the smeared dispersion coefficients interacting with themselves in Fourier space.

Aside from the energies, the computation of the virial and, especially for MD, the forces are also relevant. The equation for the virial is

$$V\mathbf{\Pi}_{\alpha\beta} = V\mathbf{\Pi}_{\alpha\beta}^{(r)} + V\mathbf{\Pi}_{\alpha\beta}^{(k)} + V\mathbf{\Pi}_{\alpha\beta}^{(0)}, \tag{2.13}$$

with

$$\begin{aligned} V\mathbf{\Pi}_{\alpha\beta}^{(r)} &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j'=1}^{N}\sum_{\mathbf{n}\in\mathbb{Z}^3} C_{ij}\left(\frac{6}{r_{ijn}^8} + \frac{6\beta^2}{r_{ijn}^6} + \frac{3\beta^4}{r_{ijn}^4} + \frac{\beta^6}{r_{ijn}^2}\right)\times \\ &\quad \exp\left(-r_{ijn}^2\beta^2\right)\mathbf{r}_{ijn,\alpha}\mathbf{r}_{ijn,\beta}, \end{aligned} \tag{2.14}$$

$$
\begin{aligned}
V\mathbf{\Pi}^{(k)}_{\alpha\beta} &= \frac{1}{2V}\sum_{\mathbf{h}\neq 0}\tilde{G}(\mathbf{h})\mathcal{S}^2(\mathbf{h}) \times \\
&\quad \left(\delta_{\alpha\beta} - \frac{3}{h^2}\frac{2b^3\sqrt{\pi}\mathrm{erfc}(b) - 2b^2e^{-b^2}}{2b^3\sqrt{\pi}\mathrm{erfc}(b) + (1-2b^2)e^{-b^2}}\mathbf{h}_\alpha\mathbf{h}_\beta\right), \qquad (2.15)\\
V\mathbf{\Pi}^{(0)}_{\alpha\beta} &= \frac{\pi^{2/3}\beta^3}{6\beta^3 V}\sum_{i,j}C_{ij}\delta_{\alpha\beta}, \qquad (2.16)
\end{aligned}
$$

where α and β are Cartesian indices and $\delta_{\alpha\beta}$ is the Dirac delta function. The expression for the force on particle i is

$$\mathbf{F}_i = \mathbf{F}_i^{(r)} + \mathbf{F}_i^{(k)}, \qquad (2.17)$$

with

$$
\begin{aligned}
\mathbf{F}_i^{(r)} &= \sum_{j'=1}^{N}\sum_{\mathbf{n}\in\mathbb{Z}^3} C_{ij}\left(\frac{6}{r_{ijn}^8} + \frac{6\beta^2}{r_{ijn}^6} + \frac{3\beta^4}{r_{ijn}^4} + \frac{\beta^6}{r_{ijn}^2}\right)\exp\left(-r_{ijn}^2\beta^2\right)\mathbf{r}_{ijn}, \qquad (2.18)\\
\mathbf{F}_i^{(k)} &= c_i\frac{1}{V}\mathrm{Im}\left(\sum_{\mathbf{h}\neq 0} ib_i\exp\left(-i\mathbf{h}\cdot\mathbf{r}_i\right)\tilde{G}(\mathbf{h})\tilde{S}(\mathbf{h})\mathbf{h}\right), \qquad (2.19)
\end{aligned}
$$

where $\tilde{S}(h)$ is the structure factor and will be addressed in more detail in Section 2.3. The superscripts in the expressions for the forces and the virial have the same meaning as for the energy above, and the truncation of the infinite sums works equivalently. Note that the self-interactions of the smeared particles have neither virial nor force contributions and that there is no force contribution from the reciprocal space term for $\mathbf{h} = 0$. To understand the benefits of the different variants of the PPPM method described later it is important to emphasize that the sum in Equation (2.19) is a reverse Fourier transform required to obtain the forces on the particles.

2.3 The structure factor and mixing rules

A requirement for the direct application of the Ewald sum is that the dispersion coefficients obey the geometric mixing rule in Equation (2.4). When this requirement is fulfilled, $\tilde{\mathcal{S}}^2(\mathbf{h})$ can be expressed as

$$\tilde{\mathcal{S}}^2(\mathbf{h}) = \tilde{S}(\mathbf{h})\tilde{S}^*(\mathbf{h}), \qquad (2.20)$$

where $\tilde{S}^*(\mathbf{h})$ is the complex conjugate of the structure factor $\tilde{S}(\mathbf{h})$, which is the continuous Fourier transform of the discrete dispersion coefficient distribution c of all particles

$$\tilde{S}(\mathbf{h}) = \sum_{j=1}^{N} c_j \exp\left(-i\mathbf{h}\cdot\mathbf{r}_j\right). \tag{2.21}$$

When the condition in Equation (2.4) is not met, $\tilde{\mathcal{S}}^2(\mathbf{h})$ has to be computed alternatively. When using a LJ potential, dispersion coefficients of unlike sites are often determined via the Lorentz-Berthelot (LB) mixing rule as

$$C_{ij} = 4\sqrt{\epsilon_i\epsilon_j}\left(\frac{\sigma_i+\sigma_j}{2}\right)^6, \tag{2.22}$$

where ϵ and σ are the LJ parameters. One approach to compute $\tilde{\mathcal{S}}^2(\mathbf{h})$ for this case is to expand the power in Equation (2.22) to arrive at [23]

$$C_{ij} = \sum_{k=0}^{6} a_{i,k}a_{i,6-k}, \tag{2.23}$$

with

$$a_{i,k} = \frac{1}{4}\sigma_i^k\sqrt{\binom{6}{k}\epsilon_i}. \tag{2.24}$$

With this splitting, $\tilde{\mathcal{S}}^2(\mathbf{h})$ can be calculated as

$$\tilde{\mathcal{S}}^2(\mathbf{h}) = \sum_{k=0}^{6} \tilde{\mathbf{S}}_k(\mathbf{h})\tilde{S}^*_{6-k}(\mathbf{h}). \tag{2.25}$$

Because of their symmetry only four of the seven addends in Equation (2.25) have to be calculated. However, the explicit computation of each of the structure factors $\tilde{S}_k(\mathbf{h})$ via

$$\tilde{S}_k(\mathbf{h}) = \sum_{j}^{N} a_{j,k} \exp\left(-i\mathbf{h}\cdot\mathbf{r}_j\right). \tag{2.26}$$

is required. On top of that, the reverse Fourier transform in Equation (2.19) has to performed for each structure factor separately when computing the forces. The usage of the LB mixing rule with the expansion of the dispersion coefficients with Equation (2.23) therefore imposes a considerable computational overhead.

In the following, we will derive an alternative approach to split the dispersion coefficients. The splitting in Equation (2.23) can be more generally written as a matrix equation:

$$\mathbf{C} = \mathbf{ADPA}^T, \tag{2.27}$$

where $\mathbf{C}$ is an $n \times n$ matrix of the dispersion coefficients, $\mathbf{A}$ is a $n \times k$ matrix with transpose $\mathbf{A}^T$ that contains the a_i, $\mathbf{D}$ is a diagonal $k \times k$ matrix and $\mathbf{P}$ is a $k \times k$ permutation matrix. Reciprocal-space computations are performed k times for this splitting. Equation (2.27) is generally applicable for splitting the dispersion coefficients in the framework of Ewald summation and is not restricted to the LB rule. When the LB mixing rule is used and dispersion coefficients are split with Equation (2.23), $k = 7$, the diagonal matrix $\mathbf{D}$ is the identity matrix, and $\mathbf{P}$ is anti-diagonal.

The computation time can be optimized by performing the matrix factorization in Equation (2.27) so that k is minimized; however, we are unaware of a general solution to this optimization problem. Thus, instead of solving Equation (2.27), we propose to split the dispersion coefficients with

$$C_{i,j} = d_0 a_{i,0} a_{j,0} + d_1 a_{i,1} a_{j,1} + ... + d_k a_{i,k} a_{j,k}, \tag{2.28}$$

which can be written in a matrix form

$$\mathbf{C} = \mathbf{ADA^T}, \tag{2.29}$$

which is Equation (2.27) when $\mathbf{P}$ is the identity matrix. Because $\mathbf{C}$ is symmetric, the matrix factorization in Equation (2.29), which is equivalent to an eigenvalue decomposition, is always possible such that $k = n$. Moreover, if $\operatorname{rank}\mathbf{C} < n$, the decomposition can be performed such that $k = \operatorname{rank}\mathbf{C}$.

The functional form of Equation (2.28) was chosen such that it can easily be used in Ewald solvers. The a coefficients are assigned to the particles, whereas the d coefficients are simply multiplied by the Green's function $\tilde{G}$. An advantage of this splitting method is that if the LB mixing rule is used, the dispersion potential is split into fewer terms if $\operatorname{rank}\mathbf{C} < 7$. This is equivalent to requiring that the number of different LJ diameters be less than 7, which is true in many simulations. In such cases, the overhead for the reciprocal-space computations is reduced. This splitting technique becomes unfavorable, though, if the number of different LJ diameters is greater than 7. This splitting method can also be applied to systems in which dispersion coefficients deviate from or do not follow any standard mixing rule, such as in the models in [53–56].

2.4 The particle-particle particle-mesh algorithm

A strong disadvantage of the Ewald method is that it requires a continuous Fourier transform to compute the structure factor: The scaling of the continuous Fourier transform is related to the amount of data that has to be transformed squared. For the case considered here, the input data is proportional to the number of atoms. The scaling of the Ewald summation with respect to the particle numbers is thus $\mathcal{O}(N^2)$, or, when exploiting symmetry, $\mathcal{O}(N^{3/2})$ [41]. As a result, this method is restricted to systems with only a few thousands of particles.

This severe restriction can be overcome by replacing the continuous Fourier transform with the discrete Fourier transform, or fast Fourier transform (FFT), which scales with $\mathcal{O}(N\log N)$ [57]. A prerequisite for the application of FFTs is that the transformed data is discretized on a regular grid, which in turn means that the particle system, in which the positions of the particles are continuous, has to be transformed to a discrete state before performing the FFT. Similarly, the results that are obtained when transforming results back to real space with the inverse FFT exist only in discrete space and have to be backinterpolated on the particles to obtain the forces. The transformation from the continuous set to a discrete set is achieved by spanning a mesh in the simulation box, onto which the dispersion coefficients are distributed. Overall, the resulting additional steps compared to Ewald summation are:

1. anterpolate the dispersion coefficients of the particles onto the grid
2. use the FFT to compute the structure factors
3. compute the interaction energies in reciprocal space
4. use the inverse FFT to transform the results back to real space
5. backinterpolate results onto the particles to obtain the forces.

The name of the grid based modification is particle-particle particle-mesh Ewald method. Expressed in equations, the only difference compared to Ewald summation method is that the reciprocal space contributions for the energy, virial, and force are computed with

$$
\begin{aligned}
E^{(k)} &= \frac{1}{2V} \sum_{\mathbf{k_n}\in\hat{\mathbb{M}},\mathbf{k_n}\neq 0} \hat{G}_{\mathrm{opt}}(\mathbf{k_n})\hat{\mathcal{S}}^2(\mathbf{k_n}), && (2.30)\\
V\mathbf{\Pi}_{\alpha\beta} &= \frac{1}{2V} \sum_{\mathbf{k_n}\in\hat{\mathbb{M}},\mathbf{k_n}\neq 0} \hat{G}_{\mathrm{opt}}(\mathbf{k_n})\hat{\mathcal{S}}_6^2(\mathbf{k_n}) \times \\
&\quad \left(\delta_{\alpha\beta} - \frac{3}{\mathbf{k_n}^2}\frac{2b^3\sqrt{\pi}\mathrm{erfc}(b) - 2b^2 e^{-b^2}}{2b^3\sqrt{\pi}\mathrm{erfc}(b) + (1-2b^2)e^{-b^2}}\mathbf{k_{n}}_\alpha \mathbf{k_{n}}_\beta\right), && (2.31)\\
\mathbf{F}_i^{(k)} &= c_i \mathbf{D}^{(k)}(\mathbf{r}_i) && (2.32)
\end{aligned}
$$

instead of Equations (2.9), (2.15), and (2.19). The modifications are (i) that the sum is only over reciprocal-space vectors $\mathbf{k_n} \in \hat{\mathbb{M}}$ on the grid $\hat{\mathbb{M}}$, (ii) that an optimized version of the Green's function $\hat{G}_{\text{opt}}(\mathbf{h})$ is used in the computation and (iii) that $\hat{\mathcal{S}}^2(\mathbf{k})$ is computed from the anterpolated set of dispersion coefficients. $\mathbf{D}^{(k)}$ is the dispersion field caused by the reciprocal space interactions and depends on the details of the method. Its computation will be described in Section 2.4.2 when these details are presented. The definition of $\hat{G}_{\text{opt}}(\mathbf{h})$ depends on these details in is therefore presented afterwards in Section 2.4.3.

In real space, the grid points onto which the dispersion coefficients are anterpolated have the coordinates $\mathbf{r}_M$ and form a rectangular grid $\mathbb{M}$. The number of grid points in each direction is N_x, N_y, and N_z. In each direction, the grid points are evenly spaced. The spacing between the grid points is $h_x = N_x/L_x$, $h_y = N_y/L_y$, and $h_z = N_z/L_z$, the grid can thus be expressed as

$$\begin{aligned}\mathbf{r_n} \in \mathbb{M} = \{&(\mathbf{n}_x/N_x)L_x\mathbf{e}_x + (\mathbf{n}_y/N_y)L_y\mathbf{e}_y + (\mathbf{n}_z/N_z)L_z\mathbf{e}_z : \\ &n_\alpha = 0, 1, \ldots, N_\alpha - 1, \alpha = x, y, z\},\end{aligned} \tag{2.33}$$

where $\mathbf{e}_x$, $\mathbf{e}_y$, and $\mathbf{e}_z$ are unit vectors in x, y, and z direction. Similarly, the reciprocal-space vectors are defined by

$$\mathbf{k_n} = 2\pi((\mathbf{n}_x/L_x)\mathbf{e}_x + (\mathbf{n}_y/L_y)\mathbf{e}_y + (\mathbf{n}_z/L_z)\mathbf{e}_z; \tag{2.34}$$

the set $\hat{\mathbb{M}}$ is defined as

$$\begin{aligned}\hat{\mathbb{M}} = \{&2\pi((\mathbf{n}_x/L_x)\mathbf{e}_x + (\mathbf{n}_y/L_y)\mathbf{e}_y + (\mathbf{n}_z/L_z)\mathbf{e}_z) : \\ &n_\alpha \in \mathbb{Z}, -N_\alpha/2 \leq n_\alpha < (N_\alpha)/2, \alpha = x, y, z\}.\end{aligned} \tag{2.35}$$

With this definition, $\mathbf{k_n} = 0$ is located close to the center of the grid, which is favorable when truncation sums over $\mathbf{k_n}$ that run beyond the extend of $\hat{\mathbb{M}}$.

A more detailed description of the different steps, as well as a description of the interlaced PPPM algorithm, a modification of PPPM targeted to obtain faster computations, is given below. Because excellent reviews for this material can already be found in the literature [34,36,50,58,59] for Coulomb interactions and because for most of the operations described above, the procedure for the dispersion case is very similar to procedures for Coulomb interactions, the method will only be presented briefly to have a self-contained description.

Before proceeding with the description of the method, a quick guide to the notation is given here: all quantities defined in continuous reciprocal space are labeled with a tilde, e.g., $\tilde{G}$ for the Green's function; quantities defined only on the grid in reciprocal space are labeled with a hat, e.g., $\hat{G}_{\text{opt}}$ for the optimized Green's function. Real-space quantities, e.g. G for the real-space version of the Green's function, do not have a special label. This notation was adopted from [50].

2.4.1 Discretization of dispersion coefficients and transformation to reciprocal space

The anterpolation of the dispersion function onto the grid points is given by

$$c_{\mathbb{M}}(\mathbf{r_n}) = \sum_{i=1}^{N} c_i W(\mathbf{r_n} - \mathbf{r}_i), \tag{2.36}$$

where $W(\mathbf{x})$ is the anterpolation function. Dispersion coefficients of particles close to the boundary of the box are wrapped around periodically when anterpolated. As with the splitting function, the choice of $W(\mathbf{x})$ is in principle arbitrary but should fulfill certain requirements, such as conservation of the dispersion coefficients, that dispersion coefficients should only be transferred to nearby grid points for computational efficiency, that the discretization error should be small, and that the function is smooth. The desirable properties of the function are discussed in more detail in [50]. The usual choice of the interpolation function by Hockney and Eastwood [34] is to obtain $W(\mathbf{x})$ from a P-fold convolution of the characteristic function $\chi_{[-1/2,1/2]}$, which is one in the interval $[-1/2, 1/2]$ and zero elsewhere, with itself. The resulting interpolation function is a polynomial of degree $P-1$; the dispersion coefficients are anterpolated on the nearest P grid points in each cell dimension with this choice of the interpolation function. P is therefore also called the interpolation order. Explicit expressions for $W(\mathbf{x})$ are given in [50] (with the difference that W in the notation used there corresponds to $W/(h_x h_y h_z)$ in the notation used here).

The dispersion coefficient density on the grid $c_M(\mathbf{r}_p)$ can then be used to compute the structure factor using the FFT:

$$\hat{S}_6(\mathbf{k_n}) = \frac{1}{h_x h_y h_z} \sum_{\mathbf{r_n} \in \mathbb{M}} c_{\mathbb{M}}(\mathbf{r_n}) \exp\left(-i\mathbf{k_n} \cdot \mathbf{r_n}\right). \tag{2.37}$$

The relation between $\hat{S}_6(\mathbf{k_n})$ and $\tilde{S}_6(\mathbf{k_n})$ can be expressed as [60]

$$\hat{S}_6(\mathbf{k_n}) = \hat{U}(\mathbf{k_n})\tilde{S}_6(\mathbf{k_n}), \tag{2.38}$$

where

$$\hat{U}(\mathbf{k_n}) = \frac{1}{h_x h_y h_z}\hat{W}(\mathbf{k_n}) = \left(\frac{\sin(\mathbf{k_n}_x h_x/2)}{\mathbf{k_n}_x h_x/2}\frac{\sin(\mathbf{k_n}_y h_y/2)}{\mathbf{k_n}_y h_y/2}\frac{\sin(\mathbf{k_n}_z h_z/2)}{\mathbf{k_n}_z h_z/2}\right)^P \tag{2.39}$$

The computation of $\hat{\mathcal{S}}^2(\mathbf{k_n})$ from the structure factor $\hat{S}_6(\mathbf{k_n})$ works similar to the computation of $\tilde{\mathcal{S}}^2(\mathbf{h})$ from $\tilde{S}_6(\mathbf{h})$ described in Section 2.3. The influence of the mixing rules is not affected.

2.4.2 Differentiation and backinterpolation

Once the dispersion potential on the grid points is available, two additional steps are required to compute the forces that act on the particles. First, since the force is the negative gradient of the energy, a differentiation of the potential is required to obtain the dispersion field $\mathbf{D}^{(k)}$. Second, the results have to be backinterpolated from the grid onto the particle positions, such that the forces can be computed with Equation (2.32). As discussed in [50], there are three different approaches to perform these operations.

The first approach is to perform the differentiation of the dispersion potential in Fourier space, which corresponds to a simple multiplication with the three-dimensional $i\mathbf{k}$-vector in reciprocal space. The obtained dispersion field is then backtransformed to real space. The forces on each particle are obtained from backinterpolating the dispersion field from the grid onto the particles. As pointed out by Hockney and Eastwood [34], to conserve momentum in the system, the backinterpolation function should be equal to the anterpolation function that was used to distribute the dispersion coefficients onto the grid. The resulting expression for the dispersion field at position $\mathbf{r}_i$ is: [61]

$$\begin{aligned}\mathbf{D}^{(k)}(\mathbf{r}_i) = \sum_{\mathbf{r_n}'\in\mathbb{M}} W(\mathbf{r}_i - \mathbf{r_n}')\frac{1}{V}\sum_{\mathbf{k_n}\in\hat{\mathbb{M}},\mathbf{k_n}\neq 0} e^{i\mathbf{k_n}\cdot\mathbf{r}_{n'}}\times \\ (-i\mathbf{k_n})\hat{G}_{\text{opt}}(\mathbf{k_n})\sum_{\mathbf{r_n}} e^{i\mathbf{k_n}\cdot\mathbf{r_n}}\sum_{j=1}^{N} W(\mathbf{r_n} - \mathbf{r}_j).\end{aligned} \tag{2.40}$$

When read from right to left, the different steps are in order as they appear when executing the algorithm. The last sum is over all particles j and is the anterpolation function. The second-to-last sum is over all grid points and is the FFT to obtain the structure factor. The second sum from the left is over all grid points in reciprocal space and is the reverse FFT of the dispersion field that has been obtained by multiplication with $(-i\mathbf{k_n})\hat{G}_{\text{opt}}(\mathbf{k_n})$. The first sum is over all grid points and performs the backinterpolation onto the particles. The $i\mathbf{k}$ differentiation has the advantage of conserving momentum and providing high accuracy in the computations, but has the disadvantage of not conserving the energy of the system [34, 50]. Moreover, because of the multiplication with the three dimensional $i\mathbf{k}$-vector, three inverse FFTs have to be performed per structure factor (cf. Section 2.3) which imposes a computational burden.

The second approach is to perform the differentiation of the dispersion potential and the backinterpolation onto the particles as a single operation. This step is performed by first doing a reverse FFT of the potential to real space and then using the derivative of the interpolation function W to compute the forces acting on each particle. The resulting expression for the dispersion field at $\mathbf{r}_i$ is [61]

$$\begin{aligned}\mathbf{D}^{(k)}(\mathbf{r}_i) = \sum_{\mathbf{r}_{\mathbf{n}'} \in \mathbb{M}} -\nabla W(\mathbf{r}_i - \mathbf{r}_{\mathbf{n}}{}') \frac{1}{V} \sum_{\mathbf{k}_{\mathbf{n}} \in \hat{\mathbb{M}}, \mathbf{k}_{\mathbf{n}} \neq 0} e^{i\mathbf{k}_{\mathbf{n}} \cdot \mathbf{r}_{n'}} \times \\ \hat{G}_{\mathrm{opt}}(\mathbf{k}_{\mathbf{n}}) \sum_{\mathbf{r}_{\mathbf{n}}} e^{i\mathbf{k}_{\mathbf{n}} \cdot \mathbf{r}_{\mathbf{n}}} \sum_{j=1}^{N} W(\mathbf{r}_{\mathbf{n}} - \mathbf{r}_j). \qquad (2.41)\end{aligned}$$

This approach is called analytic differentiation (ad). The terms in the last two sums are identical to those of the $i\mathbf{k}$ scheme. The second sum is over all reciprocal space vectors and is the reverse Fourier transform of the dispersion potential. The first sum is over all grid points and performs the backinterpolation onto the particles. This interpolation scheme uses only one reverse Fourier transform and is therefore, for a given grid size, computationally less demanding. Moreover, it has the advantage of conserving the energy of the system. The disadvantage is that momentum is not conserved because the function for the backinterpolation is not the same as that used for the anterpolation and that for a given grid size the results are less accurate than those obtained with the $i\mathbf{k}$ differentiation.

That the anterpolation and interpolation functions for the ad scheme are not identical leads to the additional disadvantage of particles having a spurious self-interaction that leads to a self-force. This self-force can be corrected by adding the self-force correction term, which for the (simplified) decoupled approximation is [62]

$$\mathbf{F}^{\mathrm{self}}_{i,\alpha} = c_i^2 \sum_{n=1}^{\infty} a_\alpha^{(n)} \sin(n 2\pi \mathbf{r}_{i,\alpha}/h_\alpha), \qquad (2.42)$$

with

$$a_\alpha^{(n)} = \frac{2\pi \mathbf{m}_\alpha}{h_\alpha} \frac{1}{V} \sum_{\mathbf{k_n} \in \hat{\mathbb{M}}, \mathbf{k_n} \neq 0} \hat{G}_{\text{opt}}(\mathbf{k_n}) \sum_{\mathbf{m'} \in \mathbb{Z}^3} \hat{U}(\mathbf{k}_{\mathbf{n}+\mathbf{Nm'}}) \hat{U}(\mathbf{k}_{\mathbf{n}+\mathbf{N}(\mathbf{m'}+\mathbf{m})}), \tag{2.43}$$

to the results obtained with Equations. (2.32) and (2.41). In Equations (2.42) and (2.43), α is a Cartesian index, $\mathbf{m}$ is set to $\mathbf{m} = n\mathbf{e}_\alpha$, $\mathbf{N}$ is a vector that contains the number of grid points N_x, N_y, and N_z in each dimension. Because the a_α coefficients decay rapidly [62], only the first 2 values of n are evaluated in the sum in Equation (2.42) in this study. $\mathbf{m'}$ is allowed to run from -2 to 2 in each dimension in the sum of Equation (2.43). A more elaborate version of this self-force correction, in which the decoupling approximation is not employed, is described in [62].

The third approach, which was not studied in this thesis, is to transform the potential field back to real space and then perform a numerical differentiation to obtain the forces. Because of the non-local character of this approach resulting from that interpolation has to be performed over a large number of grid points to obtain accurate results, which is considered a major disadvantage, this approach was not developed further.

2.4.3 The optimized Green's function $\hat{G}_{\text{opt}}$

The *ad hoc* approach to choose the Green's function for the computations on the grid $\hat{G}_{\text{opt}}$ would be to set it equal to the continuum Green's function $\tilde{G}$ in Equation (2.12). However, as shown by Deserno and Holm [50], this approach will result in low-accuracy computations compared to optimized versions of the Green's function. To minimize the error in the computed forces, the correct approach is to minimize the difference between the dispersion field obtained from an exact computation and the results that are obtained from the PPPM algorithm such that [34, 61, 63]

$$Q = \int_V \int_V \left[\mathbf{D}^{(k)}_{\text{PPPM}}(\mathbf{r}_1, \mathbf{r}_2) - \mathbf{D}^{(k)}(\mathbf{r}_1, \mathbf{r}_2) \right]^2 d^3r_1 d^3r_2 \tag{2.44}$$

is minimized, where $\mathbf{D}^{(k)}_{\text{PPPM}}(\mathbf{r}_1, \mathbf{r}_2)$ and $\mathbf{D}^{(k)}_{\text{exact}}(\mathbf{r}_1, \mathbf{r}_2)$ are the dispersion fields at position $\mathbf{r}_1$ caused by a smeared unit dispersion coefficient at location $\mathbf{r}_2$ that are obtained from the PPPM method and from exact reciprocal space computations, respectively.

As described in detail in [61, 63], Equation (2.44) can be transformed to

$$Q = \sum_{\mathbf{k_n} \in \hat{\mathbb{M}}, \mathbf{k_n} \neq 0} \left[A(\mathbf{k_n}) \hat{G}^2_{\text{opt}}(\mathbf{k_n}) - 2B(\mathbf{k_n}) \hat{G}_{\text{opt}}(\mathbf{k_n}) \right] + H, \tag{2.45}$$

with

$$H = \sum_{\mathbf{k_n} \in \hat{\mathbb{M}}, \mathbf{k_n} \neq 0} \sum_{\mathbf{m} \in \mathbb{Z}^3} \left[\tilde{G}(\mathbf{k_{n+Nm}}) \mathbf{k_{n+Nm}} \right]^2 . \tag{2.46}$$

Minimizing Equation (2.45) with respect to $\hat{G}_{\text{opt}}$ yields

$$\hat{G}_{\text{opt}}(\mathbf{k_n}) = \frac{B(\mathbf{k_n})}{A(\mathbf{k_n})} \tag{2.47}$$

for the optimal influence function $\hat{G}_{\text{opt}}$.

The different expressions for the dispersion fields $\mathbf{D}^{(k)}$ for the ad and $i\mathbf{k}$ differentiation scheme in Section 2.4.2 result in different expressions for $A(\mathbf{k_n})$ and $B(\mathbf{k_n})$. The resulting expressions are [61, 63]

$$B^{(i\mathbf{k})}(\mathbf{k_n}) = \sum_{\mathbf{m} \in \mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}}) \tilde{G}(\mathbf{k_{n+Nm}}) \mathbf{k_n} \mathbf{k_{n+Nm}}, \tag{2.48}$$

$$A^{(i\mathbf{k})}(\mathbf{k_n}) = \left[\sum_{\mathbf{m} \in \mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}}) \right]^2 \mathbf{k_n}^2 \tag{2.49}$$

for the $i\mathbf{k}$ differentiation scheme and

$$B^{(\text{ad})}(\mathbf{k_n}) = \sum_{\mathbf{m} \in \mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}}) \tilde{G}(\mathbf{k_{n+Nm}}) \mathbf{k_{n+Nm}}^2, \tag{2.50}$$

$$A^{(\text{ad})}(\mathbf{k_n}) = \left[\sum_{\mathbf{m} \in \mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}}) \right] \left[\sum_{\mathbf{m} \in \mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}}) \mathbf{k_{n+Nm}}^2 \right] \tag{2.51}$$

for the ad scheme.

An alternative approach is not to minimize the errors in the dispersion field, but in the dispersion potential [64], which results in

$$B^{(\text{eng,acc})}(\mathbf{k_n}) = \sum_{\mathbf{m} \in \mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}}) \tilde{G}(\mathbf{k_{n+Nm}}), \tag{2.52}$$

$$A^{(\text{eng,acc})}(\mathbf{k_n}) = \left[\sum_{\mathbf{m} \in \mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}}) \right]^2 , \tag{2.53}$$

$$\tag{2.54}$$

which can be approximated with

$$B^{(\text{eng})}(\mathbf{k_n}) = \hat{U}^2(\mathbf{k_n}) \tilde{G}(\mathbf{k_n}), \tag{2.55}$$

$$A^{(\mathrm{eng})}(\mathbf{k_n}) \quad = \quad \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})\right]^2 . \tag{2.56}$$

As discussed in [64], there is only a negligible difference between the different versions of the optimal influence functions computed with Equations (2.48) and (2.49) compared to Equations (2.55) and (2.56).

2.4.4 Interlacing

As shown in [63], the total error of the reciprocal space computations can be subdivided into an nonfluctuating part Q_{non} that is insensitive to the grid position, and a fluctuating part Q_{fluc} that is sensitive to shifting the position of the grid. It was furthermore shown that the second contribution can be largely decreased if the reciprocal-space computations are performed on two different grids that are shifted by half the grid spacing in each direction. The resulting force $\mathbf{F}^{(k)}_{\mathrm{int}}$ is then obtained from averaging the forces obtained from the computations on the shifted grids $\mathbf{F}^{(k)}_{\mathrm{grid1}}$ and $\mathbf{F}^{(k)}_{\mathrm{grid2}}$

$$\mathbf{F}^{(k)}_{\mathrm{int}} = \frac{\mathbf{F}^{(k)}_{\mathrm{grid1}} + \mathbf{F}^{(k)}_{\mathrm{grid2}}}{2}. \tag{2.57}$$

The disadvantage of this so-called interlaced or staggered method is obviously that the reciprocal-space computations have to be performed twice, which for a given grid dimension and interpolation order P strongly increases the computation cost. When the fluctuating error Q_{fluc} dominates the overall error in the computations, however, an increase in accuracy up to two orders of magnitude has been observed. This increase in accuracy is a promising approach to perform reciprocal space computations on a coarser grid and in this way save computation time.

Aside from having to perform the computations on the reciprocal grid twice, the main difference between interlaced and noninterlaced PPPM is that the optimal influence function changes. The expressions for the optimal influence functions of the interlaced algorithm can also be expressed with Equation (2.47). The numerators $B(\mathbf{k_n})$ remain unchanged for interlacing, whereas the denominators $A(\mathbf{k_n})$ change. The resulting expressions for the denominator are [63]

$$A^{(i\mathbf{k},\mathrm{int})}(\mathbf{k_n}) \quad = \quad (A_1^{(i\mathbf{k})}(\mathbf{k_n}) + A_2^{(i\mathbf{k})}(\mathbf{k_n}))/2, \tag{2.58}$$

$$A_1^{(i\mathbf{k})}(\mathbf{k_n}) \quad = \quad \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})\right]^2 \mathbf{k_n}^2, \tag{2.59}$$

$$A_2^{(i\mathbf{k})}(\mathbf{k_n}) \quad = \quad \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})(-1)^{\mathbf{m}_x+\mathbf{m}_y+\mathbf{m}_z}\right]^2 \mathbf{k_n}^2, \tag{2.60}$$

for $i\mathbf{k}$ differentiation,

$$\begin{aligned}
A^{(\mathrm{ad,int})}(\mathbf{k_n}) &= (A_1^{(\mathrm{ad})}(\mathbf{k_n}) + A_2^{(\mathrm{ad})}(\mathbf{k_n}))/2, &(2.61)\\
A_1^{(\mathrm{ad})}(\mathbf{k_n}) &= \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})\right]\left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})\mathbf{k_{n+Nm}}^2\right], &(2.62)\\
A_2^{(\mathrm{ad})}(\mathbf{k_n}) &= \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})(-1)^{\mathbf{m}_x+\mathbf{m}_y+\mathbf{m}_z}\right] \times &(2.63)\\
&\quad \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})\mathbf{k_{n+Nm}}^2(-1)^{\mathbf{m}_x+\mathbf{m}_y+\mathbf{m}_z}\right], &(2.64)
\end{aligned}$$

for the ad scheme, and

$$\begin{aligned}
A^{(\mathrm{eng,int})}(\mathbf{k_n}) &= (A_1^{(\mathrm{eng})}(\mathbf{k_n}) + A_2^{(\mathrm{eng})}(\mathbf{k_n}))/2, &(2.65)\\
A_1^{(\mathrm{eng})}(\mathbf{k_n}) &= \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})\right]^2, &(2.66)\\
A_2^{(\mathrm{eng})}(\mathbf{k_n}) &= \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \hat{U}^2(\mathbf{k_{n+Nm}})(-1)^{\mathbf{m}_x+\mathbf{m}_y+\mathbf{m}_z}\right]^2, &(2.67)
\end{aligned}$$

for the energy-optimized Green's function.

The spurious self-force for the ad scheme arises for each of the grids. Consequently, the self-force correction has to be applied for each of the grids separately and then averaged, similar to what is done with the forces

$$\mathbf{F}_{\mathrm{int}}^{\mathrm{self}} = \frac{\mathbf{F}_{\mathrm{grid1}}^{\mathrm{self}} + \mathbf{F}_{\mathrm{grid2}}^{\mathrm{self}}}{2}. \tag{2.68}$$

2.4.5 Error estimates

Several parameters can be tuned to influence the accuracy of the dispersion PPPM method: the chosen cutoff radius r_c for the sum in real space, the Ewald parameter β, the grid size, and the order of the interpolation function for distributing the dispersion coefficient onto a grid. The qualitative influence of the parameters can be understood easily. The real-space error arises from truncating the pair potential. Increasing the cutoff radius or the Ewald parameter, which leads to a faster decaying real-space potential, increases the accuracy in real space. The precision in reciprocal space depends on the Ewald parameter, the grid spacing, and the interpolation order. Decreasing either of the first two or increasing the latter will lead to higher accuracy in the reciprocal-space contribution.

To choose the tunable parameters effectively, a more quantitative understanding of the parameters' influence on the force error is required. This section presents estimates for the error of real- and reciprocal-space forces that can be used to set the parameters for the algorithm. The error estimates provide expressions for the root mean square (RMS) error

$$\Delta F = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\mathbf{F}_i^{\mathrm{PPPM}} - \mathbf{F}_i^{\mathrm{exact}})^2}, \tag{2.69}$$

where $\mathbf{F}_i^{\mathrm{PPPM}}$ is the force calculated with the PPPM algorithm and $\mathbf{F}_i^{\mathrm{exact}}$ is the exact result.

2.4.5.1 Real-space error estimate

The real-space error estimate is an extension of the estimate of Kolafa and Perram [65] for Coulomb interactions to r^{-6} potentials. The sum of the square of the real-space contribution of the dispersion interaction of the particles beyond the cutoff r_c on a single particle can be expressed as

$$\Delta F_i^2 \quad = \quad c_i^2 \sum_{j:r_{ij}>r_c} c_j^2 \left(\frac{6}{r_{ij}^8} + \frac{6\beta^2}{r_{ij}^6} + \frac{3\beta^4}{r_{ij}^4} + \frac{\beta^6}{r_{ij}^2}\right)^2 \exp\left(-2r_{ij}^2\beta^2\right) r_{ij}^2. \tag{2.70}$$

Assuming that the particles are randomly distributed beyond the cutoff, the sum can be replaced by an integral to arrive at

$$\begin{aligned}\Delta F_i^2 \quad &= \quad c_i^2 \sum_j c_j^2 \frac{1}{V} \int_{r_c}^{\infty} \left(\frac{6}{r_{ij}^6} + \frac{6\beta^2}{r_{ij}^4} + \frac{3\beta^4}{r_{ij}^2} + \beta^6\right)^2 \times \\ &\qquad \exp\left(-2r_{ij}^2\beta^2\right) 4\pi \mathrm{d}r.\end{aligned} \tag{2.71}$$

Using [65]

$$\int_A^{\infty} \exp(-Bx^2) f(x) \mathrm{d}x \approx \exp(-BA^2)\frac{f(A)}{2BA}, \tag{2.72}$$

we arrive at

$$\Delta F_i^2 = c_i^2 \sum_j c_j^2 \frac{\pi\beta^{10}}{V r_c} \left(\frac{6}{r_c^6 \beta^6} + \frac{6}{r_c^4 \beta^4} + \frac{3}{r_c^2 \beta^2} + 1 \right)^2 \exp\left(-2 r_c^2 \beta^2\right), \tag{2.73}$$

which leads to the averaged error in the force

$$\begin{aligned} \Delta F_{\text{real}} &= \sqrt{\frac{1}{N} \sum_i \Delta F_i^2} \\ &= \frac{\mathcal{C}\sqrt{\pi}\beta^5}{\sqrt{N V r_c}} \left(\frac{6}{r_c^6 \beta^6} + \frac{6}{r_c^4 \beta^4} + \frac{3}{r_c^2 \beta^2} + 1 \right) \exp\left(-r_c^2 \beta^2\right), \end{aligned} \tag{2.74}$$

where N is the number of particles and

$$\mathcal{C} = \sum_i c_i^2. \tag{2.75}$$

2.4.5.2 Reciprocal-space error estimate

The error in the forces in reciprocal space can be expressed by [66]

$$\Delta F_{\text{reciprocal}} = \mathcal{C} \sqrt{\frac{Q}{NV}}, \tag{2.76}$$

where Q is defined in Equation (2.44). Equation (2.76) is based on the assumption that the error on a specific particle arises from the interactions with all other particles, that the error is proportional to the interaction strength between the particles, and that the error contributions arising from different sites are uncorrelated.

2.5 Numerical tests: accuracy and single-core performance

In this section we perform numerical tests of the algorithm presented above. In particular, we will examine the accuracy that can be obtained with the different variants of PPPM and how well the error estimates perform. Furthermore, the single-core performance of the different variants will be compared. Finally, as the expected main benefit of the PPPM algorithm over the plain Ewald method is the improved scaling behavior with respect to the number of particles, we perform a comparison of the timing of these methods.

2.5.1 Accuracy of variants of the PPPM method

We performed test runs to examine the accuracy of the different versions of PPPM for dispersion and the error estimates presented in Section 2.4.5. We randomly placed 2000 LJ particles with energy ϵ and diameter σ in a box of length $15\,\sigma$ in each direction to create a bulk system. To test the error estimates for surface systems, we placed 5200 LJ particles randomly in a $30\sigma \times 30\sigma \times 10\sigma$ box and extended the length of the shortest box edge to $30\,\sigma$ afterwards without changing the particle coordinates. In addition to the systems in which the particles have random positions, we equilibrate the systems above for 10 000 timesteps in the NVT ensemble at $T = 1.2\epsilon/k_B$, where k_B is the Boltzmann constant; the surface system is equilibrated before the box is extended. In this way the particle distributions have a stronger connection to a real system in which particles will be more structured than in a random state. States with strong particle overlaps, which can occur for systems where the particles are simply positioned randomly, are avoided in these equilibrated states.

We calculated the real and reciprocal space forces on the particles for these configurations separately using different values for the Ewald parameter, the grid size, the interpolation order, and the real space cutoff. $k = 2, \ldots, 6$ mesh points were used in each direction. Interpolation orders $P = 3, \ldots, 6$ were examined but results are reported for $P = 5$ only because no additional qualitative insight can be obtained from other interpolation orders. Real-space cutoffs of 2.0, 2.5, 3.0, 3.5 and $4.0\,\sigma$ were used. The RMS error in the forces is calculated with Equation (2.69) for the real-space and the reciprocal-space separately. The "exact" reference force $\mathbf{F}_i^{\mathrm{exact}}$ is computed with an Ewald summation [23] in which we used a large cutoff and a large number of reciprocal vectors to ensure proper convergence.

Figure 2.3 shows the reciprocal space accuracy that is obtained by using the $i\mathbf{k}$ or ad schemes and different influence functions. For each PPPM variant, we used the optimal influence function that is force-optimized for the given scheme, i.e. Equations (2.48) to (2.51) and (2.58) to (2.64), or the energy-optimized function, i.e. Equations. (2.55) to (2.56) and (2.65) to (2.67). There are two main findings from this figure. First, the well-known fact that the $i\mathbf{k}$ differentiation usually provides more accurate results than the ad scheme [50] is reproduced. Exceptions to this trend are observed in regions of low accuracy. Second, the energy-optimized version of the influence function $\hat{G}_{\mathrm{opt}}$ provides accuracy that is practically indistinguishable from what is observed with the influence functions that are optimized with respect to the differentiation scheme. The only exception to this behavior is the staggered ad scheme. The bump that is obtained with the ad-optimized influence function is not reproduced with the energy-optimized version of

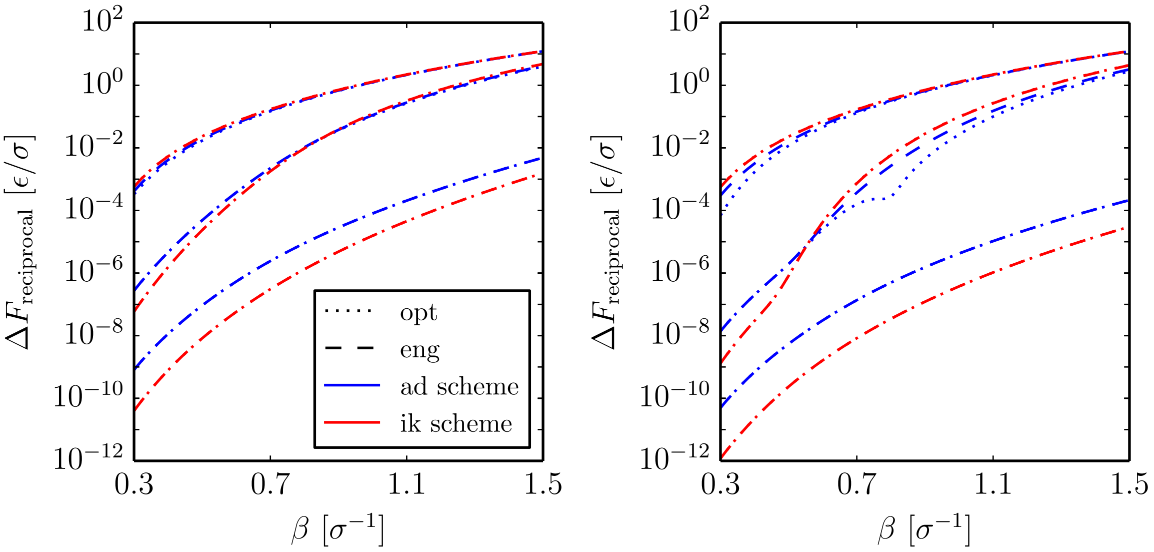


Figure 2.3: Force errors of the different versions of PPPM for dispersion. Left: normal PPPM; right: interlaced version. From top to bottom the number of grid points in each direction increases as 4, 16, 64. Dotted lines are for versions of $\hat{G}_{\text{opt}}$ optimized for $i\mathbf{k}$ or ad differentiation. Dashed lines are for energy optimized Green's functions. The line color represents the differentiation scheme.

the influence function. As a result, computations with the ad-optimized influence function will provide better results when the simulations are run at the conditions where the bump is located. The bump is where the different contributions Q_{non} and Q_{fluc} have similar contributions to the error [63] and can best be seen in the computations with a grid size of 16 points in each dimension. Because the energy-optimized influence function provides comparable accuracy as the differentiation-specific optimized functions, because Equation (2.55) can be computed with less computational effort, and because analytic expressions are available for Equations (2.56) and (2.65) [67], this version of the Green's functions is preferred. The only exception is the interlaced ad scheme, in which using the ad optimized Green's function can be favorable in terms of accuracy.

Differences in the accuracy that are obtained when applying the algorithm to randomly distributed particles or equilibrated systems are shown in Figure 2.4. Although the image shows results obtained with the non-interlaced $i\mathbf{k}$ scheme, conclusions are transferable to all other test cases. Force errors are always smaller in the equilibrated system. The difference is especially pronounced in regions of low accuracy. The reason is that strongest errors arise from particles that are closely spaced: extremely closely spaced particles cannot occur in equilibrated systems, whereas they can if the particles are randomly distributed.

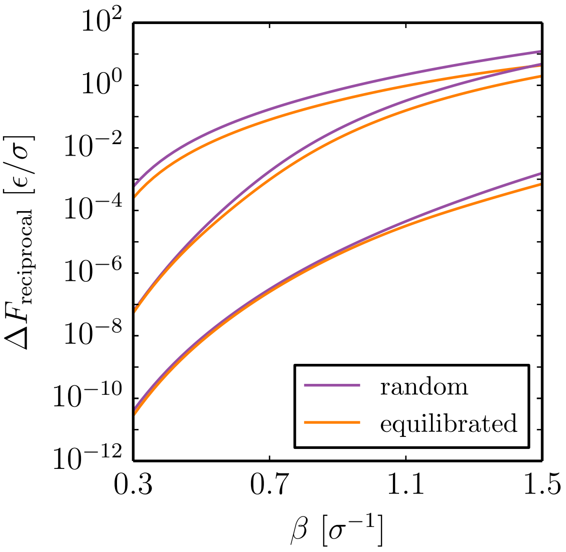


Figure 2.4: Accuracy of PPPM for random particle positions and equilibrated systems for $i\mathbf{k}$ differentiation. The number of grid points in each direction increases as 4, 16, 64 from top to bottom. The accuracy is higher for equilibrated systems at high force errors.

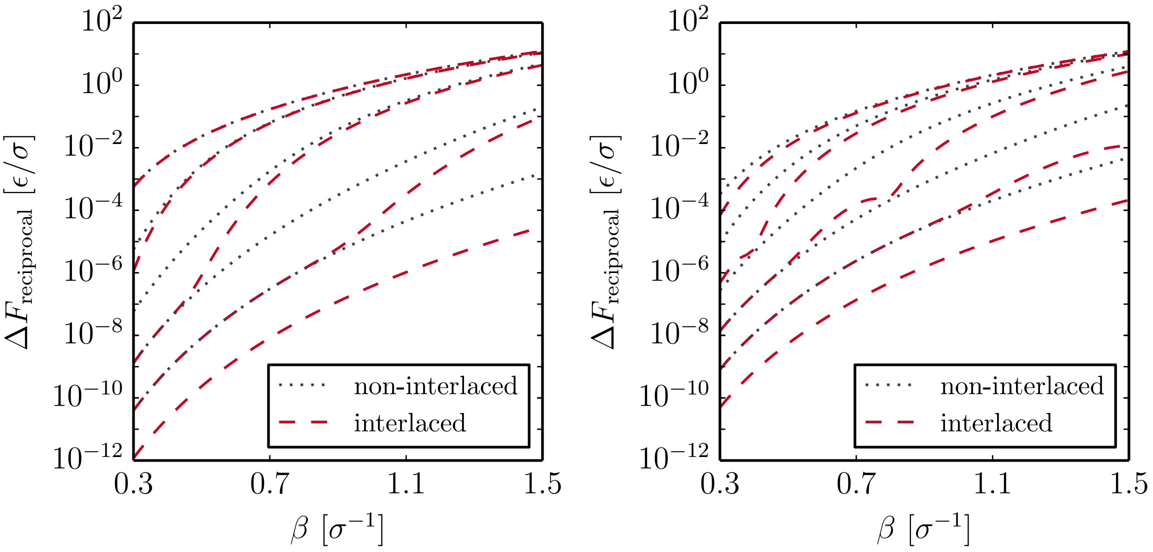


Figure 2.5: Comparison of the accuracy obtained with normal and interlaced PPPM. Left: $i\mathbf{k}$ differentiation; right: ad scheme, each with specific optimized influence functions. The number of grid points in each direction increases as 4, 8, 16, 32, 64 from top to bottom. For large force errors, the interlaced version of the algorithm provides almost identical accuracy as non-interlaced PPPM. For small force errors, the interlaced version can be as accurate as the non-interlaced version with halved grid spacing. (The uppermost of the blue lines approximately matches the uppermost of the red lines, whereas the lowermost of the blue lines matches the second-lowest of the red lines.)

Differences between the interlaced and non-interlaced version of the algorithm are depicted in Figure 2.5. As expected, the interlaced version of the algorithm always provides increased accuracy for a given grid size. It is important to note, however, that the increase in accuracy is strongly dependent on the state of the simulations. In regions of low accuracy, the results obtained with the interlaced algorithm are almost indistinguishable from the non-interlaced version for a given grid size. Using interlacing in this case will not provide any benefit in the accuracy but will only yield a penalty in computation times. In regions of high accuracy, the interlaced version of the PPPM algorithm can provide accuracy that is almost identical to the accuracy obtained with the non-interlaced version of the code with a halved grid spacing, i.e., 8 times more grid points. This gain in accuracy is promising for increasing the performance when a specific accuracy is required. These issues are further addressed in Sections 2.5.3 and 2.9.

2.5.2 Accuracy of error estimates

The test systems for the computations presented here are identical to those used in Section 2.5.1. The results of the real-space error estimate are given in Figure 2.6, the reciprocal-space error estimates are shown for the randomized bulk and surface system in Figure 2.7 and in Figure 2.8. Except for small values of β, the real-space error estimate works well for the bulk system. The error is slightly underestimated in surface simulations. For the reciprocal-space error estimate, the same behavior is observed. The error estimates can thus be used to select PPPM parameters to achieve a desired real- and reciprocal-space accuracy. As a final comment, we note that while for the force computation using Equations. (2.55) and (2.56) to compute the optimized Green's function $\hat{G}_{\mathrm{opt}}$ works well, the usage of the energy-optimized expressions for A and B is of course unqualified for computing the error estimates.

2.5.3 Performance comparison of variants of the PPPM method

The performance of the different variants of the PPPM method, namely the $i\mathbf{k}$ and ad differentiation schemes and the interlaced and non-interlaced versions, were computed as a function of force accuracy. A test system was prepared by creating a cubic box with length 10σ in which 4000 LJ particles with interaction parameters $\epsilon = 1$ and $\sigma = 1$ were placed at random position. The system was allowed to equilibrate for 10 000 timesteps of 0.005τ in the NVT ensemble using a Nosé-Hoover thermostat [68] with coupling constant $\tau_T = 0.5\tau$. Afterwards the system was replicated in each dimension by a

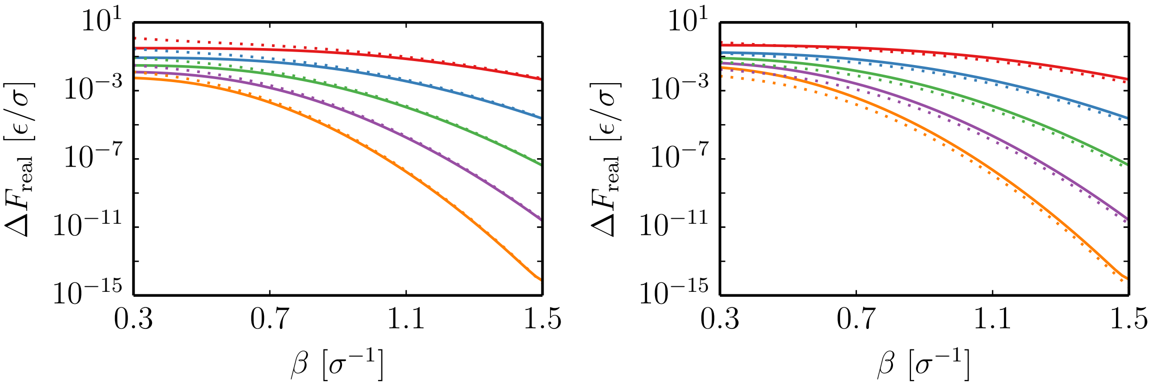


Figure 2.6: Comparison of the real-space error estimate with measured force errors. Left: equilibrated bulk system; right: interfacial system. Solid line: measured error, dotted line: estimated error. The cutoff radius increases from top to bottom. Overall, the estimate is too pessimistic for small values of β and too optimistic for the interfacial system.

factor of 4 such that the final system contains 256 000 particles and has a box length of 40σ. The system was then run 10 times for 100 timesteps in the NVE ensemble for each set of examined parameters on a single core of the Intel IvyBridge EP processor to obtain average times per timestep t_{step} and mean uncertainties.

For the cutoff we chose $r_c = 3\sigma$ and as interpolation order we chose $P = 5$ in all computations. The Ewald parameter β and grid spacing are determined based on the error estimates in Section 2.4.5 by specifying a target accuracy for the equal real- and reciprocal-space computations. We first set the Ewald parameter to achieve the desired real-space computation. Once the Ewald parameter is fixed, the grid spacing is the only free variable, which is then set to achieve the desired reciprocal-space accuracy. Afterwards, the Ewald parameter is adjusted to balance the error between the real- and reciprocal space accuracy. The resulting accuracy will thus be slightly different compared to the accuracy that was initially specified. The overall accuracy is estimated as [66]

$$\Delta F = \sqrt{\Delta F^2_{\mathrm{real}} + \Delta F^2_{\mathrm{reciprocal}}}. \tag{2.77}$$

The timing data over the force error is given in Figure 2.9 for the different variants of the PPPM algorithm. As expected, all variants follow the trend that computations with increasing accuracy (i.e., small values of ΔF) are increasingly expensive. For large errors in the computations, the non-interlaced versions of the algorithm provide better performance than the interlaced version, which is in agreement with the interlaced version of the algorithm not increasing the accuracy in regions of large ΔF, as shown above. In regions of small force errors, however, the interlaced version strongly

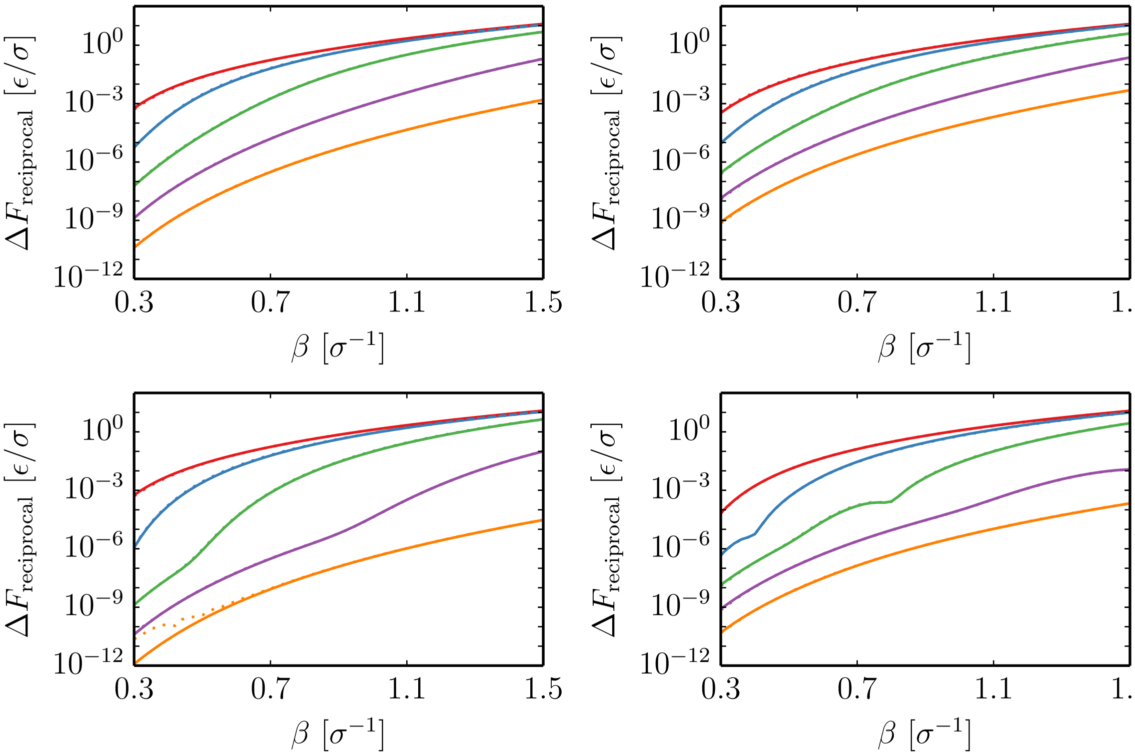


Figure 2.7: Comparison of the reciprocal-space error estimate with measured force errors in the bulk system with random particle positions. Top and bottom: normal and interlaced PPPM. Left and right: $i\mathbf{k}$ and ad scheme. The number of grid points in each direction increases from top to bottom as 4, 8, 16, 32, 64 in each image. Solid line: measured error, dotted line: estimated error. The predictions of the error estimate are in excellent agreement with the measurements. Deviations are only observed for the interlaced $i\mathbf{k}$ version at extremely high accuracy and are possibly a result of numeric errors.

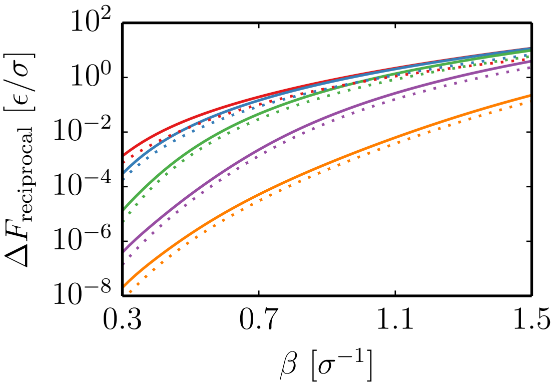


Figure 2.8: Reciprocal space estimate compared to measured force errors in the random interfacial system for the non-interlaced ad version of the algorithm. The error estimate is slightly too pessimistic. $P = 5$ and the number of grid points increases from top to bottom as 4, 8, 16, 32, 64.

outperforms the non-interlaced version. It is noteworthy that the gain of the interlaced version is much larger in regions of high force accuracy than its loss in regions of low force accuracy. For example, the interlaced ad version of the algorithm at an error of $\Delta F \approx 0.0002\epsilon/\sigma$ is around a factor of 4 faster than the non-interlaced versions. In the regions of low force accuracy, the increase in computation time is only around 30 %. The observed trend is in agreement with results for the PPPM method for dipolar interactions [69].

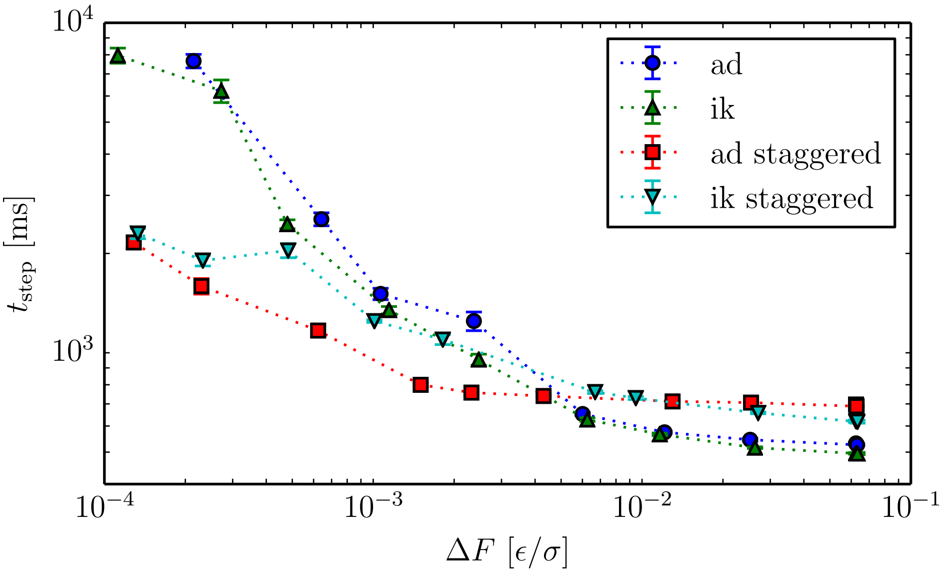


Figure 2.9: Single-core timing per timestep of PPPM variants versus the force accuracy.

A more detailed comparison shows that the interlaced ad scheme is fastest for high force accuracy and slowest for low force accuracy When comparing the ad and the $i\mathbf{k}$ methods, a crossover point can be found for the interlaced versions of the $i\mathbf{k}$ and the ad scheme, whereas for the noninterlaced version the $i\mathbf{k}$ scheme is always faster. When performing this detailed comparison one needs to consider, however: (i) that the performance is strongly dependent on the implementation and the trends for these details necessarily must be attributed to implementation details; (ii) that the PPPM parameters were not determined from a rigorous optimization over the entire parameter space, so the selected parameters might be especially well-chosen for one case whereas they are particularly bad for the other; (iii) that all computations were performed on a single core which means that the heavy communication burden that can arise on multi-core machines for performing the 3D FFTs is not considered. These more detailed differences between the interpolation schemes will thus possibly depend on the exact choice of parameters, the system size, the architecture on which the simulation is run, and of course the details of the implementation.

2.5.4 Scaling behavior compared to Ewald summation

The main benefit of mesh-based Ewald methods over traditional Ewald sums is the improved scaling behavior of the mesh-based approach. To examine the scalability of the implemented solver we have performed simulations with $2^n \times 10^3$ LJ particles, where $n = 0, 1, \dots, 10$, with the dispersion PPPM solver and the Ewald summation. The density was $3.64\sigma^{-3}$ in all simulations. The boxes were always cubic. An energy minimization and equilibration over 50 000 timesteps in the NVT ensemble at $T = 0.85\epsilon/k_B$ was followed by a simulation over 1 000 timesteps in the NVE ensemble. The simulation time of the last 1 000 timesteps was used to measure the performance. These simulations were executed on a single core of an Intel Harpertown E5454 processor with eight 3.0 GHz Xeon cores.

Automated parameter generation was applied in simulations with the Ewald sum [23]. For the PPPM algorithm we use the $i\mathbf{k}$ scheme without interlacing. The parameters are set using the same approach described in Section 2.5.3 with a desired real- and reciprocal-space error of $0.01\,\epsilon/\sigma$.

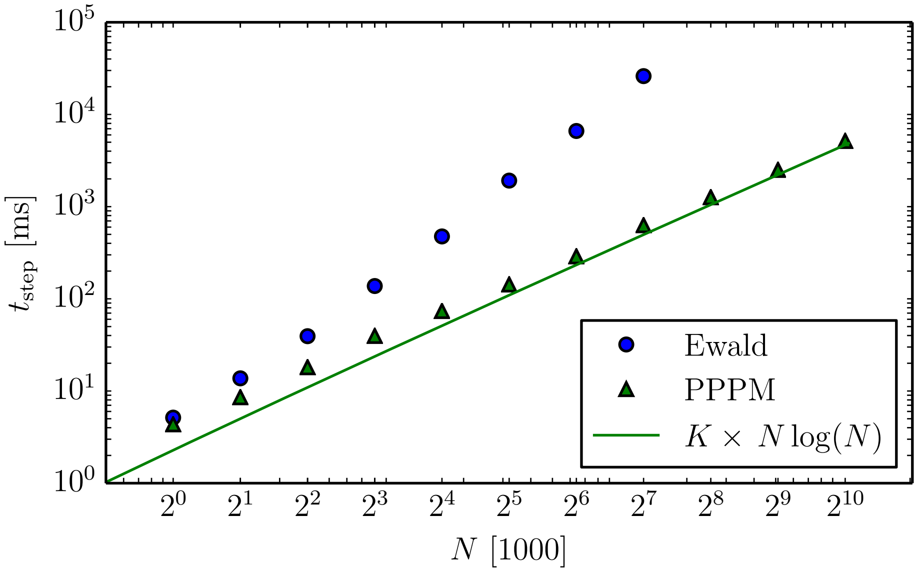


Figure 2.10: Comparison of the scaling of the Ewald method and PPPM with increasing number of particles. The PPPM algorithm follows the expected $\mathcal{O}(N \log N)$ scaling and outperforms the Ewald method for high particle numbers by orders of magnitude. (Modified reprint with permission from Ref. [70])

As can be seen from Figure 2.10, which shows the computation time per timestep, the dispersion PPPM method approaches the expected scaling behavior of $\mathcal{O}(N \log N)$ with increasing numbers of particles. Its performance becomes several orders of magnitude faster than the traditional Ewald sum and is thus far more suitable for large-scale computations. The comparison between the different solvers drawn here should be considered qualitative, as we did not examine whether the two different solvers were run with the same accuracy.

2.6 Impact of the real- and reciprocal-space errors on physical behavior

Using PPPM introduces a set of parameters which strongly influence the accuracy and the efficiency of a simulation. As often observed in computational methods, an increase in accuracy will result in a performance penalty. The target when selecting parameters for PPPM will typically be to achieve maximum performance while having sufficient accuracy that the obtained physical results are not biased by inaccurate force computations. This target can only be achieved with a proper understanding of how inaccuracy affects the physical behavior of a simulated system. In this section we will first perform a numerical study with the model system hexane to address these issues. Hexane was chosen as a model system because dispersion interactions dominate and errors in the computation of dispersion interactions will have a strong and therefore easily noticeable effect on the results. Afterwards we will provide a theoretical analysis of the results and comment how the real- and reciprocal-space errors should be selected.

2.6.1 Parameter study with hexane

Hexane was modeled using the OPLS-AA [27] force field. Simulations contained 689 hexane molecules that were placed using PACKMOL [71] in a subvolume around the center of the box with volume $50\times50\times150\,\text{Å}^3$. After an energy minimization with a soft potential and several runs with restricted movement of the particles, the simulations were equilibrated for 1 000 000 timesteps with a timestep of $\Delta t = 1\,\text{fs}$. The temperature was set to $T = 300\,\text{K}$ using a Nosé-Hoover [68] thermostat with a damping factor of 0.1 ps. Afterwards, simulations were run for 5 ns and surface tensions and liquid densities were determined as described in Appendix A. Instantaneous surface tensions were computed every timestep, particle positions to compute the density profiles were stored every 12.5 ps. PPPM [34] with a real space cutoff of $r_c = 10\,\text{Å}$, an Ewald parameter of $\beta_1 = 0.17\,\text{Å}^{-1}$ and fifth-order interpolation $P = 5$ was used to calculate the electrostatic potential. The grid dimension was set to $20 \times 20 \times 45$.

The parameters of the PPPM method for dispersion are the real space cutoff, the

Ewald parameter, the interpolation order, and the grid spacing in each dimension. The influence of the different parameters on the force accuracy is already described at the beginning of Section 2.4.5. Instead of exploring this six-dimensional parameter space, we set the interpolation order to $P = 5$ and the real space cutoff $r_c = 10.0\,\text{Å}$. This choice of parameters was made because these values are commonly used in MD simulations, although they are in principle arbitrary. Furthermore, the grid spacing was equal in all three dimensions because near-cubic grids usually provide most accurate calculations.

This reduction of the parameter space allows for determining suitable simulation parameters with less effort, but permits reaching a wide range of accuracy in either real or reciprocal space. As the real-space cutoff is fixed, the real-space accuracy depends only on the Ewald parameter, which is used in the following simulations to tune the real-space accuracy. In principle, we could also have fixed the Ewald parameter beforehand and modified the real-space cutoff to tune the real-space accuracy, but we decided against it to have better control over the real-space calculation time. For a given Ewald parameter and the other parameters fixed, the grid spacing can be altered to tune the reciprocal-space accuracy.

We calculated the surface tension and liquid density with different settings for the two remaining parameters, the Ewald parameter and the uniform grid spacing. In addition, we determined the RMS error in the total forces as well as the real- and reciprocal-space contributions to the error by comparing the forces calculated for a single snapshot of an equilibrated system to forces that were calculated using a large real-space cutoff and a very small grid spacing.

The results for the surface tension and density of hexane are given as a function of the total RMS error in the forces in Figure 2.11. In simulations with fewer grid points, the total error is always dominated by the reciprocal-space error. In simulations with smaller grid spacing $h = 50/24\,\text{Å}$, the real- and reciprocal-space error are approximately equal for the highest achieved total accuracy at $\beta_6 = 0.28\,\text{Å}^{-1}$. As can be seen from Figure 2.12, the real-space error dominates for smaller values of the Ewald parameter β_6, whereas the reciprocal-space error dominates for larger values of β_6.

As the total error decreases, the simulated surface tensions and densities plateau, indicating that further increases in accuracy, which can be obtained by using even finer grids and larger values for the Ewald parameter, will offer little benefit in the accuracy of the measured quantities. Decreasing the Ewald parameter, thereby increasing the real-space error, strongly influences the simulated quantities. In contrast, increasing the Ewald parameter and in this way increasing the reciprocal-space error has less influence on the results. Physical data begin to change for reciprocal-space errors above approximately $0.01\,\text{kcal mol}^{-1}\,\text{Å}^{-1}$. For the examined quantities, the real-space

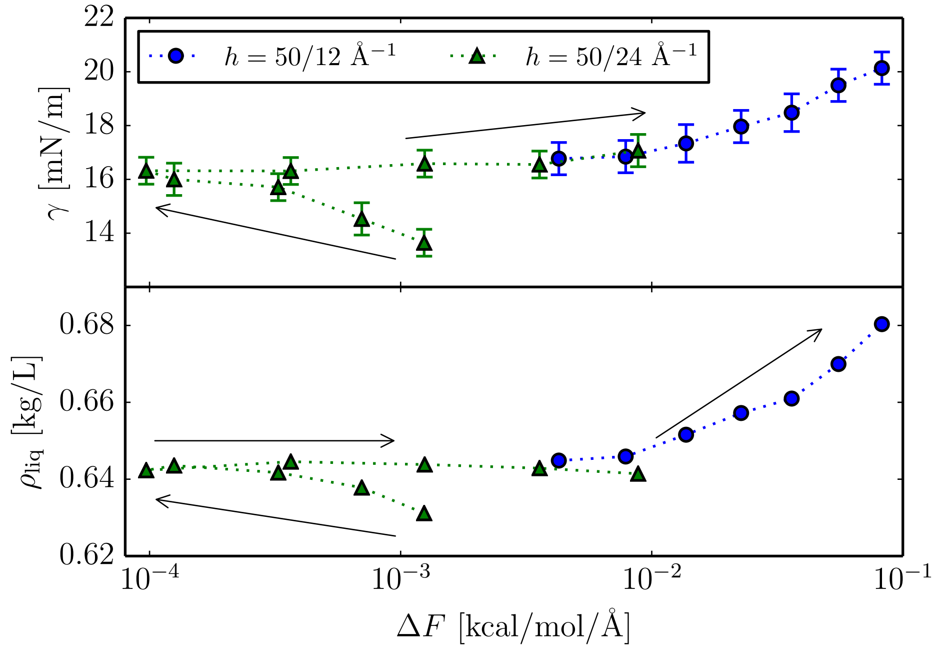


Figure 2.11: Surface tension and density of hexane as a function of the total error in the calculated forces. The arrows point in the direction of increasing Ewald parameter. The values plateau at low force error. (Modified reprint with permission from Ref. [70])

error apparently has a stronger influence on the results than the reciprocal-space error. The reason for this observation will be addressed in Section 2.6.2.

The data given in Figure 2.11 is also given on the left side of Figure 2.12 as a function of the Ewald parameter. These results, in combination with those from Figure 2.11, show that an Ewald parameter of approximately $\beta_6 = 0.28\,\text{Å}^{-1}$ in combination with a real-space cutoff $r_c = 10\,\text{Å}$ provides sufficient real-space accuracy for the performed simulations.

As the results from Figure 2.11 indicate that increasing the reciprocal-space error does not alter the obtained physical data strongly, we have performed further simulations with fixed Ewald parameter and varying grid spacing. Results of these simulations are given on the right side of Figure 2.12. Decreasing the grid spacing below $h = 50/12\,\text{Å}$ does not alter either the simulated density or surface tension, although the error in the forces continues to decrease. Because of the extended running times with higher meshes, these higher-fidelity calculations are computationally undesirable.

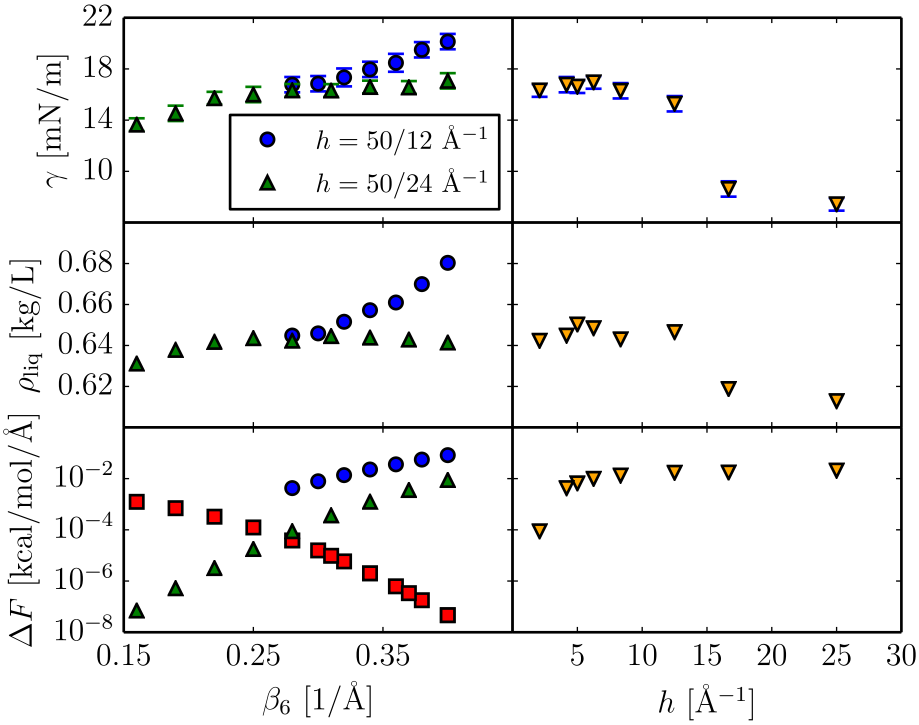


Figure 2.12: Surface tension, density, and errors in the forces in simulations of hexane. In the lower left graph, the red squares correspond to the real-space error, while the triangles and circles correspond to the reciprocal-space error when using the fine grid and coarse grids, respectively. The triangles in the lower right graph correspond to the reciprocal-space error. The Ewald parameter is $\beta_6 = 0.28\,\text{Å}^{-1}$ in all figures on the right side. (Modified reprint with permission from Ref. [70])

2.6.2 Analysis of the errors

For Coulomb interactions, it is well-established to minimize the overall RMS error by tuning the Ewald parameter so that the real- and reciprocal-space errors are approximately equal [50]. While performing this minimization is also possible for dispersion interactions, the results reported in the previous Section 2.6.1 suggest that the real-space error has a stronger impact on the physics than the reciprocal-space error. This observation will be analyzed here. It will be shown that this observation is not coincidental but has a physical basis. We show that this is the major difference between PPPM for Coulomb and dispersion interactions and needs to be taken into account when selecting PPPM parameters.

The origin for the observed difference is that electrostatic interactions between particles can be either attractive or repulsive because they can be charged negatively or positively. In contrast, dispersion interactions between pairs of particles are always attractive. This difference has important implications when truncating interactions at the real-space cutoff. For electrostatic interactions, the neglected interactions outside the cutoff can cause attractive as well as repulsive forces on a given particle. However, since dispersion interactions are always attractive, neglected interactions beyond the real-space cutoff always reduce the cohesion of a system. Concretely, this has the effect that if no correction methods are applied, the truncation of the real-space part of the dispersion potential leads to increased energy and pressure of a system.

The effect of truncating dispersion interactions on the forces is of less concern for simulations of homogeneous bulk systems, as error cancellation works well for these systems because neglected attractive interactions in one direction partly cancel out against neglected attractive interactions that would arise from interactions with particles on the opposite side. This is also the reason why, in contrast to energy and pressure corrections, force corrections are not required to account for the truncation of dispersion interactions in homogeneous bulk-phase systems.

For inhomogeneous systems, especially systems that contain an interface between a condensed phase and a vapor phase, the situation is different, however, as illustrated by Figure 2.13. The image on the left side, which shows the surface of a hexane simulation, can be transformed with a bit of abstraction to the image on the right, where the left half is the condensed phase and the right half a vapor phase. The attractive forces exerted by the particles in the condensed phase outside the cutoff radius on a particle at the surface are not counterbalanced by neglected forces from the vapor phase. As a result, the neglected force contribution for that particle caused by dispersion interactions points toward the interface. In contrast, Coulomb interactions from the condensed phase have attractive as well as repulsive forces on the particle at the interface. While the errors caused by truncating the dispersion potential accumulate, the errors caused by truncating the Coulomb interactions even partly cancel for particles located at the interface.

This effect was quantified using a snapshot of an equilibrated slab system of 5000 SPC/E water molecules [72] in a $50 \times 50 \times 150\,\text{Å}^3$ box. The system was equilibrated for 1 ns in the *NVT* ensemble using a Nosé-Hoover thermostat [68] at a temperature of 300 K. The slab has surface normals parallel to the z-direction and is located in the center of the simulation box. For this snapshot, we computed the number density in z-direction, as well as different types of force errors averaged over slabs parallel to the interface of the system. The force errors that we determined are (i) the truncation error of the dispersion forces when using a simple cutoff scheme with $r_c = 23\,\text{Å}$; (ii) the real- and reciprocal-space error of the dispersion forces when using PPPM for dispersion with a cutoff $r_c = 7\,\text{Å}$, interpolation order $P = 5$, an Ewald parameter

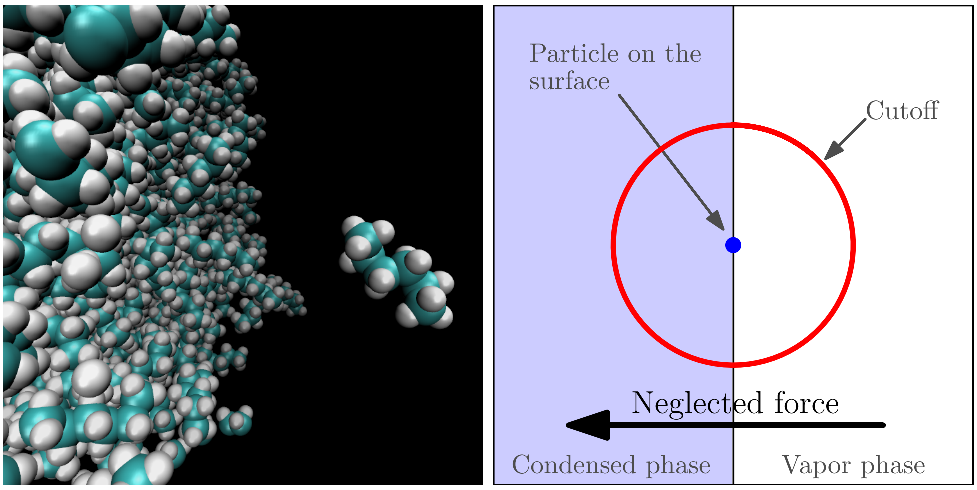


Figure 2.13: Left: Hexane surface obtained from simulation. Right: abstracted image with the focus on a particle at the surface. The error caused by truncating the interactions with particles in the condensed phase is not counterbalanced because of the void space at the right. The truncation error is therefore directed and not random.

$\beta_6 = 0.4\,\text{Å}^{-}1$, a grid spacing of $h \approx 1.67\,\text{Å}$ and the ad scheme; (iii) the real- and reciprocal-space errors of the Coulomb forces when using PPPM for electrostatics with the same cutoff and interpolation order, an Ewald parameter $\beta_1 = 0.375\,\text{Å}^{-1}$, a grid spacing of $h \approx 1.25\,\text{Å}$ and the ad scheme. The errors averaged over slabs in z-direction were computed from a comparison with a force computation performed with PPPM for Coulomb and dispersion interactions with very high accuracy.

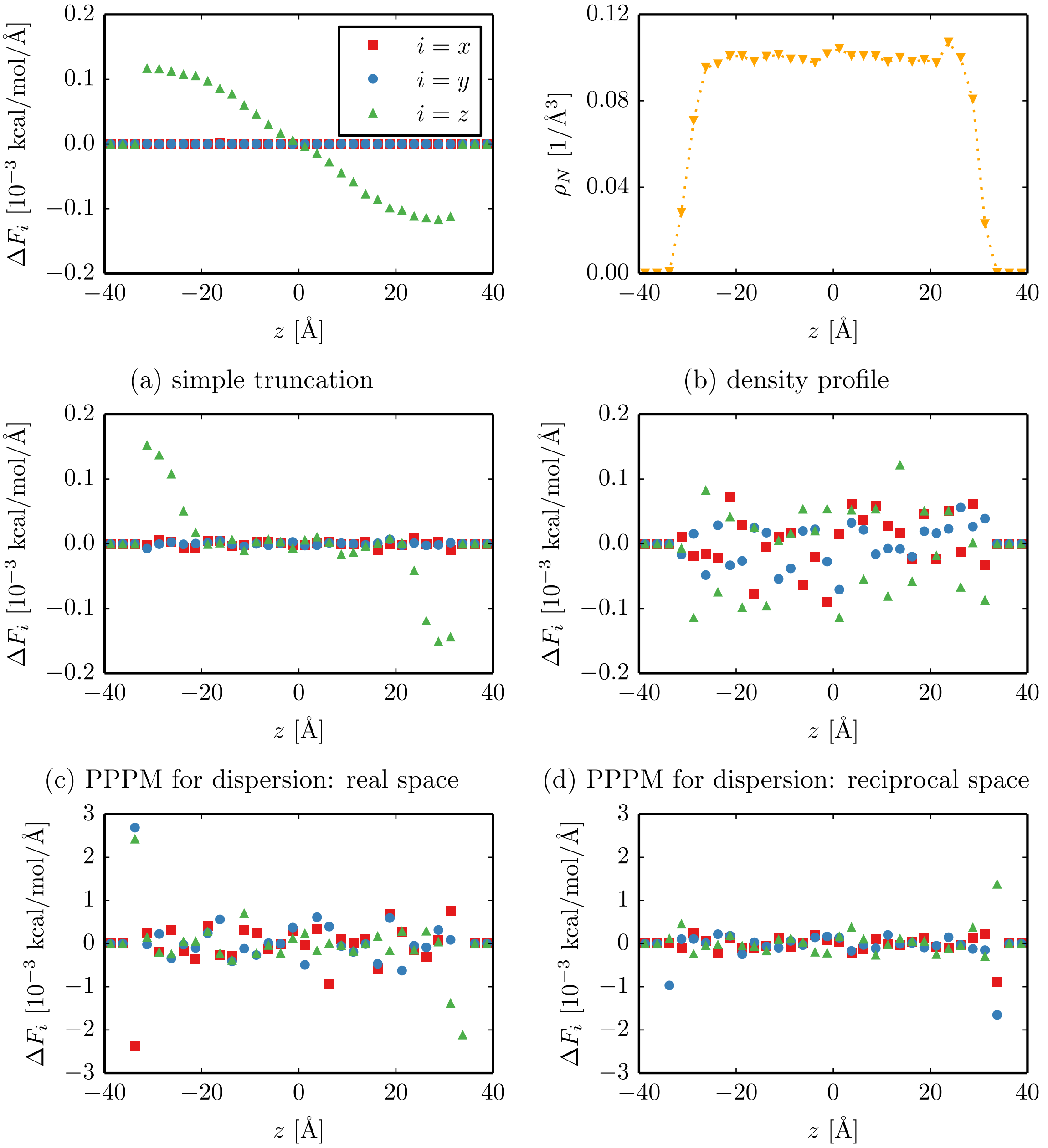
The results for the measured histograms of the density and force errors are shown in Figure 2.14. The difference between the average error in the forces ΔF_x and ΔF_y for the real-space contribution from dispersion interactions in Figure 2.14c in the x and y-direction parallel to the surface is strikingly different compared to the error ΔF_z in z-direction. ΔF_x and ΔF_y are approximately constant over the entire range of the bulk fluid system. In contrast, the absolute value of ΔF_z behaves similar to ΔF_x and ΔF_y away from the interface but strongly grows near the interface. Moreover, ΔF_z has positive values at the interface for low values of z and negative values for the interface located at high values of z. This shows that the neglected force contribution points towards the condensed phase, which is a confirmation of the accumulation of errors described above.

Comparing the error in the real-space dispersion forces to that obtained from the reciprocal-space forces in Figure 2.14d reveals that this accumulation only occurs for the real-space part. For the reciprocal-space part, the error does not have a clear direction but has the character of noise for all Cartesian dimensions in the system.

Because of the more statistical behavior of this error, the reciprocal-space error has less impact on physical results compared to the directed real-space error and is therefore more tolerable.

The error in the dispersion forces caused by a simple truncation of the potential is shown in Figure 2.14a. ΔF_x , ΔF_y, and ΔF_z behave similarly to what is observed for the truncation of the real-space dispersion potential in Figure 2.14c. The most important difference here is that while the directedness of the real-space error vanishes a few Ångstroms away from the interface when using PPPM, the directedness of ΔF_z persists in the entire liquid slab for the simple truncation scheme and slowly transforms from the positive extreme at the interface located at low values of z to the negative extreme at the interface located at high values of z.

The error in the Coulomb forces has stochastic behavior over the entire region of fluid in the simulation cell for both the real-space and the reciprocal-space error, as shown in Figures 2.14e and 2.14f. The only exceptions are the strong outliers that occur at the outermost position of the interface on both sides. These outliers result of poor statistics. Averages in the force error are here taken over only two and four atoms, respectively. As already described above, errors in the computation of Coulomb interactions can also cancel in real space. Therefore both the real-space and the reciprocal-space errors have noisy characters and can be set to similar values to obtain accurate and efficient simulations with the Coulomb PPPM.



(a) simple truncation

(b) density profile

(c) PPPM for dispersion: real space

(d) PPPM for dispersion: reciprocal space

(e) PPPM for electrostatics: real space

(f) PPPM for electrostatics: reciprocal space

Figure 2.14: Upper right: density profile of the examined surface systems. Other images: measured local force errors subdivided into the different spatial dimensions. The different error types are given in the subcaptions. For the truncation error or real-space error of dispersion, the force error is directed towards the interface for the z dimension, whereas it is more random for all other interactions.

2.7 Influence of the PPPM method on physical properties

In this section we outline how physical quantities are affected by inclusion of long-range dispersion interactions with the PPPM algorithm compared to simple trunction. For this purpose we have performed slab simulations with systems of LJ particles, SPC/E water [72], and hexane modeled with the OPLS all-atom force field [27]. These systems cover a model system as well as realistic systems in which Coulomb interactions (water) and dispersion interactions (hexane) dominate. Furthermore, these systems have already been studied and allow comparison to results from the literature [23, 45, 73–75]. A comparison with results from Shi et al. [49] is of special interest, as they have also used a PPPM dispersion method to determine the surface tension of SPC/E water.

2.7.1 Lennard-Jones particles

The LJ simulations were performed in a box with volume $11.01\sigma \times 11.01\sigma \times 176.16\sigma$ and 4000 particles that were placed randomly in a subvolume at the center of the box. After minimization using a soft potential, the system was equilibrated for 100 000 timesteps. The timestep was set to $0.005\,\tau$, where $\tau = \sigma\sqrt{m/\epsilon}$. Simulations were executed at reduced temperatures $T \in \{0.7, 0.85, 1.1, 1.2\}\epsilon/k_B$ using a Nosé-Hoover [68] thermostat with damping factor $10\,\tau$. The equations of motion were solved using a velocity Verlet algorithm [76]. Afterwards, simulations were run for another 1 000 000 timesteps with the same conditions and analyzed as described in Appendix A. During that time, instantaneous surface tensions were calculated every timestep. Configurations were stored every 1 000 timesteps to calculate the density profile. For simulations without a long-range dispersion solver, we examined cutoffs of 2.5σ, 5σ, and 7.5σ. In simulations with dispersion PPPM we used cutoffs of 3σ, 4σ, and 5σ. We used $P = 5$, $\beta = 1.1\sigma^{-1}$ and a grid with $9 \times 9 \times 144$ mesh points. These parameters were determined from numerical tests similar to those performed for the hexane system described in Section 2.6.1.

Results are given in Table 2.1. Overall, we find good agreement with results from the literature [23, 74]. The simulated densities and surface tensions show a strong dependence on the chosen cutoff in simulation without a long-range dispersion solver. The effect of the size of the cutoff is strongest for simulations at higher temperatures. Systems with small cutoffs were so close to the critical point that error functions were no longer appropriate for describing the density profile, as can be seen from

				Surface tension $[\epsilon\sigma^{-2}]$		
T $[\epsilon/k_B]$	solver	r_c $[\sigma]$	ρ_{liq} $[\sigma^{-3}]$	γ_p	γ_t	γ
0.7	cutoff	2.5	0.7865	0.588(30)	0.327	0.915(30)
		5.0	0.8349	1.006(30)	0.125	1.131(30)
		7.5	0.8390	1.112(30)	0.057	1.169(30)
	PPPM	3.0	0.8404	1.158(30)	-	1.158(30)
		4.0	0.8407	1.167(30)	-	1.167(30)
		5.0	0.8408	1.157(30)	-	1.157(30)
0.85	cutoff	2.5	0.6996	0.341(22)	0.221	0.562(22)
		5.0	0.7672	0.700(26)	0.098	0.798(26)
		7.5	0.7730	0.781(32)	0.046	0.827(32)
	PPPM	3.0	0.7748	0.817(26)	-	0.817(26)
		4.0	0.7758	0.829(24)	-	0.829(24)
		5.0	0.7756	0.829(28)	-	0.829(28)
1.1	cutoff	2.5	n.a.	0.023(26)	n.a.	0.023(26)
		5.0	0.6282	0.278(26)	0.042	0.320(26)
		7.5	0.6385	0.293(24)	0.026	0.319(24)
	PPPM	3.0	0.6451	0.314(24)	-	0.314(24)
		4.0	0.6448	0.330(26)	-	0.330(26)
		5.0	0.6462	0.302(22)	-	0.302(22)
1.2	cutoff	2.5	n.a.	0.001(20)	n.a.	0.001(20)
		5.0	0.5613	0.113(20)	0.025	0.138(20)
		7.5	0.5725	0.159(26)	0.013	0.172(26)
	PPPM	3.0	0.5767	0.164(24)	-	0.164(24)
		4.0	0.5757	0.155(22)	-	0.155(22)
		5.0	0.5766	0.154(26)	-	0.154(26)

Table 2.1: Simulated densities and surface tensions for the LJ system. Statistical uncertainties are given in parentheses. (Modified reprint with permission from Ref. [70])

Figure 2.15, whereas a stable liquid phase was formed when using larger cutoffs or PPPM for dispersion. Agreement between simulated data with and without long-range dispersion solver can only be obtained when using a large cutoff in simulations without the long-range solver. Unlike the simulations with simple truncation, the results for the dispersion PPPM method do not show a dependence on the real-space cutoff.

2.7.2 SPC/E water

Simulations with SPC/E water were performed with 5 000 water molecules in a box of 50 Å×50 Å×150 Å. The initial configurations of the particles were created using

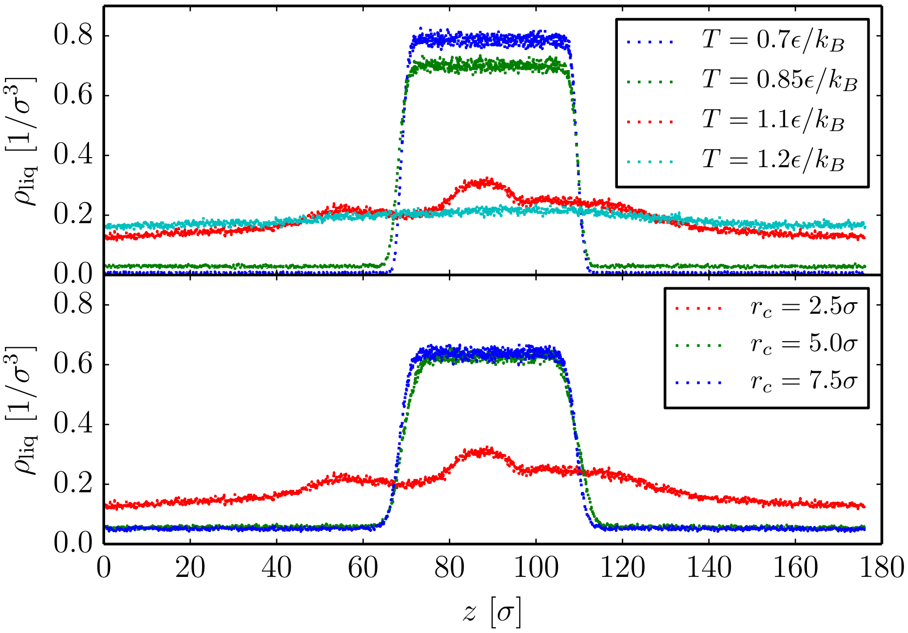


Figure 2.15: Measured density profiles for simulations of LJ particles: (top) simulations with cutoff $r_c = 2.5\sigma$ and (bottom) simulations at $T = 1.1\epsilon/k_B$. (Modified reprint with permission from Ref. [70])

PACKMOL [71]. If not explicitly given in the following, the simulation was run and analyzed as the hexane simulation in Section 2.6.1.

Simulations were executed at 300, 350, and 400 K. We examined cutoffs of 10, 12, and 16 Å for dispersion and Coulomb interaction. The SHAKE algorithm [77] was used to constrain the bond lengths and bond angles. A PPPM [34] solver was used for long-range electrostatics. We picked interpolation order $P = 5$ and a grid of $24 \times 24 \times 54$ mesh points as grid parameters and used $i\mathbf{k}$ differentiation. The Ewald parameter was $\beta_1 = 0.255$, 0.226, and 0.184 Å^{-1} for the three different cutoffs. In simulations with PPPM solver for dispersion, we used interpolation order $P = 5$. Following our results from Section 2.6.1, a grid with $12 \times 12 \times 36$ mesh points was used for the dispersion interactions. The Ewald parameter for dispersion was set to $\beta_6 = 0.28$ Å^{-1} for all cutoffs.

Table 2.2 shows the results of the simulations. When not using a long-range solver, the simulated density shows a slight dependence on the chosen cutoff radius, whereas practically no dependence can be observed when using a long-range solver for dispersion. For simulated surface tensions, neither the cutoff nor the chosen dispersion solver have a strong influence. The weak or non-existent influence on physical properties is due to the fact that Coulomb, and not dispersion, interactions are the dominant

				Surface tension [mN/m]		
T [K]	solver	r_c [Å]	ρ_{liq} [kg/L]	γ_p	γ_t	γ
300	cutoff	10.0	0.9882	54.59(100)	5.27	59.86(100)
		12.0	0.9918	56.38(100)	3.72	60.10(100)
		16.0	0.9944	57.51(84)	2.12	59.63(84)
	PPPM	10.0	0.9965	60.72(90)	-	60.72(90)
		12.0	0.9964	60.11(80)	-	60.11(80)
		16.0	0.9963	59.64(90)	-	59.64(90)
		10.0	0.9964	61.06(80)	-	61.06(80) †
350	cutoff	10.0	0.9539	47.40(60)	4.81	52.21(60)
		12.0	0.9576	48.71(60)	3.42	52.13(60)
		16.0	0.9607	49.78(70)	1.96	51.74(70)
	PPPM	10.0	0.9629	53.29(60)	-	53.29(60)
		12.0	0.9631	52.35(70)	-	52.35(70)
		16.0	0.9630	52.30(70)	-	52.30(70)
400	cutoff	10.0	0.9067	39.89(60)	4.20	44.09(60)
		12.0	0.9114	40.42(60)	3.02	43.44(60)
		16.0	0.9151	41.36(58)	1.75	43.11(58)
	PPPM	10.0	0.9177	43.98(60)	-	43.98(60)
		12.0	0.9178	43.89(60)	-	43.89(60)
		16.0	0.9178	43.44(60)	-	43.44(60)

Table 2.2: Simulated densities and surface tensions for the SPC/E system. Statistical uncertainties are given in parentheses. Simulation labeled with † was run at increased accuracy. (Modified reprint with permission from Ref. [70])

contribution to the interactions in this system.

Again, our results are in good agreement with the majority of the literature [23, 73–75]; however, they differ substantially from those reported by Shi et al. [49], who performed simulations of SPC/E with PPPM for dispersion, too. For example, their result for the surface tension at 300 K is more than 70 mN/m (read from Fig. 6 in [49]), whereas the surface tensions in our simulations are always about 60 mN/m, consistent with other studies. To ensure the validity of our results we have run an additional simulation with increased accuracy, in which we set the Ewald parameter for dispersion to $\beta = 0.3$ Å^{-1}, the interpolation order to $P = 5$ and the grid spacing to $h \approx 1.56$ Å corresponding to $32 \times 32 \times 96$ mesh points. Results of this simulation, marked with a dagger in Table 2.2, are in good agreement with the rest of our results. The increased value for the surface tension in simulations by Shi et al. might be related to the small number of water molecules (800) in their simulation or the choice of the Ewald parameter (0.9, units not given), but is most likely caused by their short sampling time of only 100 000 timesteps, as substantially longer run times are required to achieve equilibration for water at an interface [78].

2.7.3 Hexane

If not explicitly stated in the following, all settings for the hexane simulations were as reported in Section 2.6.1. We studied temperatures of 300, 350, and 400 K and cutoffs of 10, 12, and 16 Å. A PPPM solver was used for electrostatics. The grid size was set to 20 × 20 × 45 and the interpolation order to $P = 5$. The Ewald parameter was approximately $\beta_1 = 0.17$, 0.16, and 0.14 Å^{-1} for the different cutoffs. In simulations with PPPM for dispersion, the interpolation order was set to $P = 5$, the grid size was set to $12 \times 12 \times 36$ and the Ewald parameter was set to $\beta_6 = 0.28$ Å^{-1} in all simulations.

				Surface tension [mN/m]		
T [K]	solver	r_c [Å]	ρ_{liq} [kg/L]	γ_p	γ_t	γ
300	cutoff	10.0	0.6058	7.83(50)	4.59	12.42(50)
		12.0	0.6251	10.21(50)	3.75	13.96(50)
		16.0	0.6367	13.00(50)	2.34	15.34(50)
	PPPM	10.0	0.6434	16.41(50)	-	16.41(50)
		12.0	0.6439	16.16(50)	-	16.16(50)
		16.0	0.6453	15.89(50)	-	15.89(50)
350	cutoff	10.0	0.5237	2.18(40)	2.13	4.31(40)
		12.0	0.5534	4.78(40)	2.20	6.98(40)
		16.0	0.5721	7.44(50)	1.71	9.15(50)
	PPPM	10.0	0.5823	9.97(44)	-	9.97(44)
		12.0	0.5839	9.77(60)	-	9.77(60)
		16.0	0.5851	9.89(44)	-	9.89(44)
400	cutoff	10.0	n.a	-1.55(32)	n.a.	-1.55(32)
		12.0	0.4467	0.30(36)	0.74	1.04(36)
		16.0	0.4905	1.83(36)	0.85	2.68(36)
	PPPM	10.0	0.5099	4.59(36)	-	4.59(36)
		12.0	0.5106	4.66(40)	-	4.66(40)
		16.0	0.5157	4.45(46)	-	4.45(46)

Table 2.3: Simulated densities and surface tensions for the hexane system. Statistical uncertainties are given in parentheses. (Modified reprint with permission from Ref. [70])

The results are summarized in Table 2.3. The chosen cutoff radius has a strong influence on the results in simulations without a long-range dispersion solver. In contrast, the results for the PPPM method show no dependence on the chosen cutoff radius.

Our results for the PPPM algorithm are in good agreement with those from Ismail et al. [45] in simulations with an Ewald sum for dispersion. This, and the fact that the chosen cutoff does not influence the results, confirms the validity of our simulations and the good choice of the Ewald and grid parameters. As can be seen from Figure 2.16, the simulation results when not using a long-range dispersion solver approach those obtained with PPPM when increasing the cutoff. However, even those with a cutoff of 16 Å provide surface tensions and densities that are below those obtained from PPPM simulations. The incorporation of long-range dispersion solvers is thus especially important in systems in which dispersion interactions are dominant.

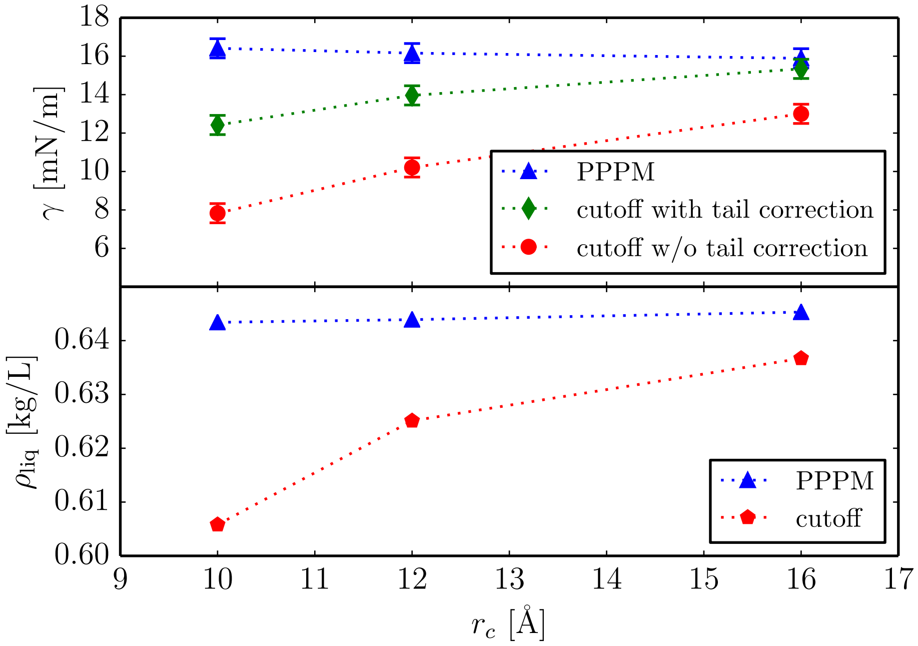


Figure 2.16: Surface tensions and densities simulated when using PPPM or a plain cutoff for dispersion. Results obtained when not using a long-range solver strongly depend on the chosen cutoff and approach the results of the simulation with PPPM with increasing cutoff size. (Modified reprint with permission from Ref. [70])

2.8 Increased performance by using long-range dispersion solvers

As shown above, mesh-based Ewald summation methods combine the advantages of the methods mentioned in Section 2.1: their accuracy can be tuned, they are applicable to arbitrary periodic systems, and their $\mathcal{O}(N \log N)$ scaling enables large-scale simulations. Moreover, mesh-based Ewald methods for dispersion interactions are similar to their Coulomb interactions counterparts. Consequently, adding dispersion PPPM solvers to codes that already include either the PPPM [34] or the PME [35,36] method should be straightforward.

Many of the advantages of mesh-based dispersion solvers are obvious from theoretical considerations. Despite that, relatively limited development of these solvers occurred prior to this study [36, 49]. To some extent, the development of these solvers has not been pushed forward earlier because homogeneous correction methods may be applicable and because adding mesh-based Ewald methods to a code requires significant effort although, as mentioned above, it is certainly manageable, especially if codes already have a mesh-based Coulomb solver. The most pressing criticism of mesh-based Ewald dispersion solvers is the belief that the required additional computations will impose an unacceptably high computational overhead.

We show that the assumption that mesh-based dispersion solvers inevitably lead to an increase in computing time is flawed, and that simulations with the PPPM method for dispersion can actually be both faster and more accurate than using a truncated potential. The performance gain compared to simulations in which dispersion interactions are simply truncated emerges from the simulation cutoff becoming a tuning parameter for long-range dispersion solvers.

In Section 2.8.1 we present the idea of using the cutoff as a tuning parameter. We also show limits of the applicability of the cutoff as a parameter for tuning the efficiency and provide a set of parameters for the PPPM method that leads to both accurate results and efficient simulations. These parameters have been used in large-scale performance tests described in Section 2.8.2. We show that simulations with the PPPM method for dispersion can outperform simulations in which dispersion interactions are truncated. We discuss our findings in Section 2.8.3.

2.8.1 Performance tuning with flexible choice of cutoffs

2.8.1.1 Theoretical considerations in the usage of smaller cutoffs

The error introduced by truncating the real-space term is related to the area of the real-space potential beyond the cutoff. This error can be decreased by increasing either the cutoff radius r_c or the Ewald parameter β_p, as indicated in Figure 2.17. This allows for more flexible choice of cutoffs, since for any r_c, the Ewald parameter can be adjusted to achieve a given accuracy in the real-space computations. Because of the strong influence of the cutoff on the computation time, smaller cutoffs facilitated by the use of an Ewald method are a promising way to accelerate simulations.

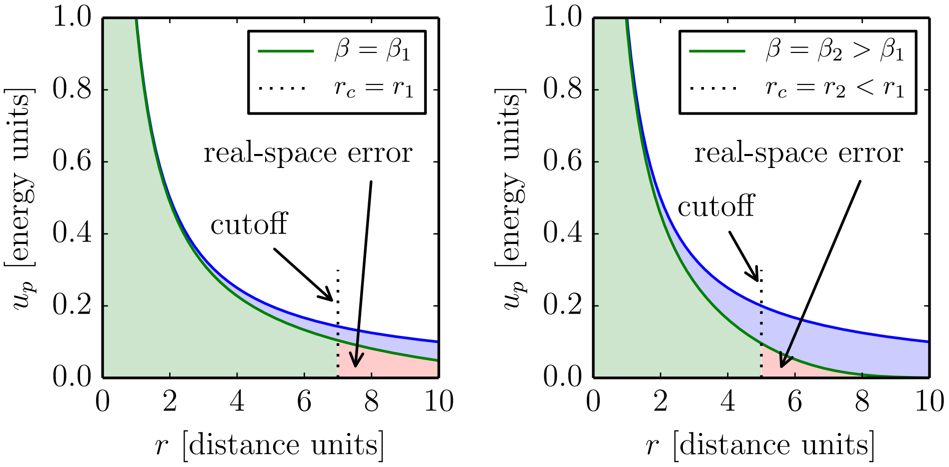


Figure 2.17: Schematic representation of the impact of the Ewald parameter β and cutoff r_c on the real-space error. The effect on the real-space error of decreasing the cutoff r_c from the left to the right image is counterbalanced by increasing the Ewald parameter.

While a desired real-space accuracy can be obtained for any choice of the cutoff radius r_c by adjusting the Ewald parameter, there are two different reasons why there is a lower bound for simulation cutoffs, even when using the Ewald method. First, as the cutoff radius decreases, the Ewald parameter must increase to achieve the same real-space accuracy [60, 66]. Increasing the Ewald parameter, however, simultaneously reduces the accuracy of the reciprocal-space calculations, which can be corrected by increasing either the interpolation order or the number of grid points, both of which increase the computation time. Below a certain cutoff threshold, the Ewald parameter becomes so large that sufficient reciprocal-space accuracy cannot be achieved for any reasonable choice of the number of grid points and interpolation order.

The second reason for a lower limit of the cutoff arises because nonbonded potentials are usually composed of different terms. In such cases, the lower limit for the cutoff is determined by the truncation error of the slowest-decaying potential for which no long-range solver is applied. For example, consider the frequently used combination of Coulomb and LJ interactions

$$u_{\mathrm{nb},ij} = \frac{q_i q_j}{r_{ij}} - \frac{C_{ij}}{r_{ij}^6} + \frac{D_{ij}}{r_{ij}^{12}}, \tag{2.78}$$

where q_i and q_j are the partial charges of particles i and j, r_{ij} is the distance between them, and $C_{ij} = 4\epsilon_{ij}\sigma_{ij}^6$ and $D_{ij} = 4\epsilon_{ij}\sigma_{ij}^{12}$ are LJ coefficients. The need for including a long-range solver for the Coulomb part of this potential is generally accepted. When using a mesh-based Ewald solver, for any cutoff radius r_c there exists an Ewald parameter β_1 that can achieve a desired accuracy for the real-space Coulomb term.

The error caused by truncating the LJ potential at r_c, however, cannot be compensated for if no long-range solver is used for the second and third terms in Equation (2.78). In this case, the r^{-6} dispersion term in Equation (2.78) controls the error. As a consequence the lower value for acceptable cutoffs is typically in the range from 10 Å to 14 Å depending on the desired accuracy and the selected force field in simulations of homogeneous systems or slightly inhomogeneous systems [27, 79, 80]. It is worth mentioning that the trend in molecular simulations has largely been moving towards using larger cutoffs [26]. The performance gain from using small cutoffs is counterbalanced by the truncation error of the r^{-6} term.

If an Ewald-based solver is also applied to the r^{-6} dispersion term in Equation (2.78), the rapidly decaying repulsive term (here r^{-12}) becomes the slowest-decaying potential, and its truncation error determines the lower limit of r_c. This facilitates decreasing the computation time by using smaller cutoffs without loss of accuracy. As shown below, making the cutoff radius r_c a tuning parameter when using a PPPM dispersion solver can lead to a net increase in performance and accuracy compared to simulations in which dispersion interactions are simply truncated.

2.8.1.2 Numerical tests with smaller cutoffs

We have performed a series of interfacial simulations with the OPLS-AA hexane [27] system to determine the lower limits for the cutoff when using a long-range dispersion solver. Except for the parameters for the long-range dispersion solver, simulations were performed as in Section 2.6.1. For the dispersion interactions, we used real-space cutoffs of $r_c \in \{6, 7, 8, 9\}$ Å. The uniform grid spacing and ranges of Ewald parameters used in these simulations is given in Table 2.4. The cutoff for the repulsive part of the LJ potential was equal to the cutoff used for the dispersion part.

r_c [Å]	h_6 [Å]	β_6 [Å^{-1}]
6	50/96	0.48 to 0.6
7	50/64	0.34 to 0.48
8	50/48	0.25 to 0.42
9	50/40	0.2 to 0.37

Table 2.4: PPPM parameters in simulations to find the lower bound for the cutoff. (Reprinted with permission from Ref. [81])

The simulations were analyzed as described in Appendix A. The averages of the computed densities are reported in Figure 2.18. The plotted error bars are twice the computed standard error in the mean in each direction. Results from Section 2.6.1

with a cutoff of 10 Å are also included in the figure. Simulated densities are given as a function of the average error in the forces. This error was determined from a single, well-equilibrated configuration as described above.

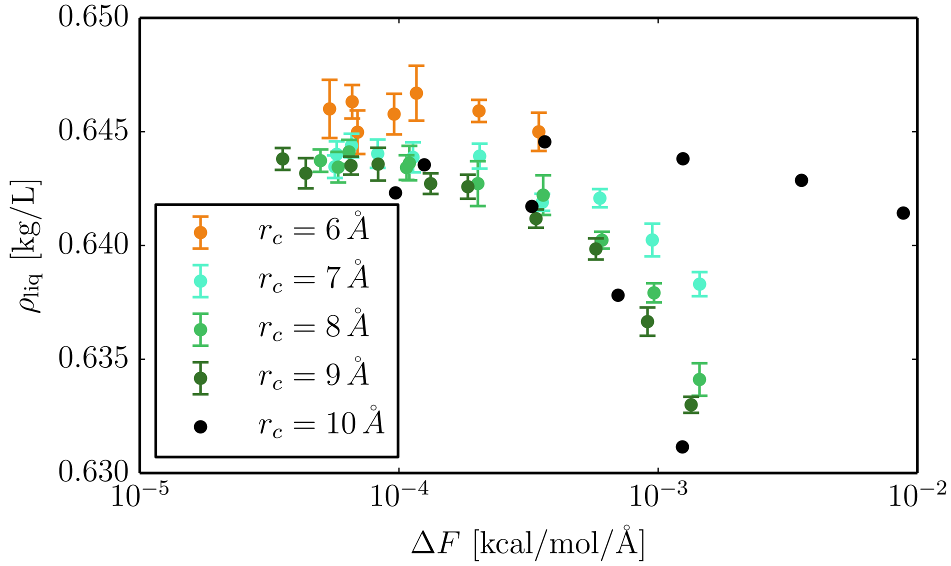


Figure 2.18: Simulated densities with different cutoffs r_c for the LJ potential and different Ewald parameters β_6 as a function of the measured error in the forces. Data for $r_c = 10$ Å was taken from Section 2.6.1. Results for $r_c = 6$ Å are different from those obtained with larger cutoffs. (Modified reprint with permission from Ref. [81])

Figure 2.18 shows that, as the accuracy of computed dispersion forces increases, the simulated densities reach a plateau for each of the chosen cutoff radii. For cutoffs $r_c \in \{7, 8, 9, 10\}$ Å, this plateau is located at approximately 0.643 kg/L. For a cutoff radius of $r_c = 6$ Å, however, the plateau is slightly higher at approximately 0.646 kg/L. The increased simulated density arises because the particles feel significantly less repulsion for this small cutoff, indicating that errors caused by truncating the repulsive part of the LJ potential begin to affect the behavior of the system. For cutoffs $r_c \geq 7$ Å, however, simulated densities are indistinguishable from results obtained in simulations with larger cutoff radii in the plateau region in Figure 2.18.

Errors in the forces ΔF_{rep} that arise from truncating the repulsive potential have been determined similar to Equation (2.69). For a cutoff of $r_c = 7$ Å the error is approximately $\Delta F_{\text{rep}} \approx 0.00015\,\text{kcal}\,\text{mol}^{-1}\,\text{Å}^{-1}$. For a cutoff of $r_c = 6$ Å, the measured error is approximately $\Delta F_{\text{rep}} \approx 0.0009\,\text{kcal}\,\text{mol}^{-1}\,\text{Å}^{-1}$. As the minimum possible cutoff is determined from the maximum LJ diameter ($\sigma_{\text{CC}} = 3.5$ Å in these simulations), we deduce that cutoffs down to twice the maximum LJ diameter are applicable when using long-range dispersion solvers without influencing the results by truncating the repulsion term.

2.8.1.3 Parameter selection for optimal accuracy and efficiency

In this section we present a set of parameters for PPPM for dispersion and briefly review the employed methods for accelerating the simulations with which increased accuracy and efficiency can be obtained. The proof that the techniques and parameters used provide accurate results is given in this section, whereas the performance benefit is demonstrated in Section 2.8.2.

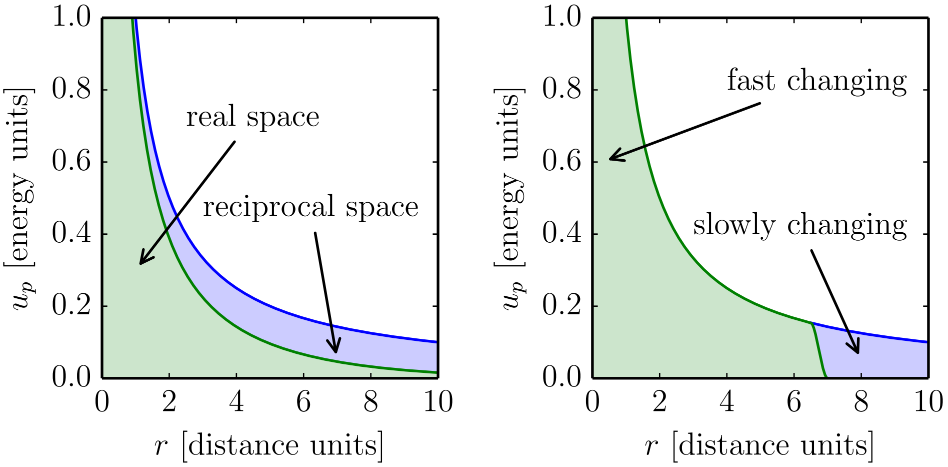


Figure 2.19: Splitting of a pair potential u_p without (left) and with rRESPA (right). Without rRESPA, real- and reciprocal-space contributions are computed with the same frequency. Using rRESPA it is possible to exploit the less-rapidly changing interactions between particles at large separations and compute this slowly changing part less frequently. Neither real- nor reciprocal-space errors are affected by the different splitting approaches. (Modified reprint with permission from Ref. [81])

As maximum performance for the reciprocal-space computations can be obtained using $i\mathbf{k}$ or analytic (ad) differentiation depending on the simulations settings [66], we provide parameters for both options. Self-force correction has been applied to Coulomb and dispersion forces in simulations with ad differentiation [62]. To decrease the effort for the real-space computations we used the double-precision extraction method of Wolff and Rudd [82]. Moreover, we have employed the reversible reference system propagator algorithm [83] (rRESPA), which uses a different force splitting which requires less frequent computation of the reciprocal-space forces (cf. Figure 2.19) [84]. Usage of rRESPA reduces the effort for the reciprocal space calculations by roughly n_r, the ratio of how often the fast-changing part of the potential is evaluated compared to the slowly changing part. We carried out rRESPA simulations with $n_r = 4$. The fast-changing part of the potential was smoothly shifted to zero between 6.5 Å and 7.0 Å and evaluated every femtosecond.

PPPM parameters for Coulomb and dispersion interactions were chosen such that the simulation results are not influenced by inaccuracy in the PPPM method. Parameters for the Coulomb interactions were generated by applying LAMMPS routines based on error estimates summarized in [60] to a bulk sample of water with a desired accuracy of $\Delta F \approx 0.03\,\mathrm{kcal\,mol^{-1}\,\AA^{-1}}$. Dispersion parameters were chosen such that, for the hexane system, the real-space error was $\Delta F \approx 0.0002\,\mathrm{kcal\,mol^{-1}\,\AA^{-1}}$, which is the threshold above which the real-space error influences the simulation results. This can be seen from Figure 2.18, considering that with the PPPM parameters used in Section 2.8.1.2 the real-space error is approximately equal to the total error at the point where simulated densities start to decrease. The reciprocal-space error was $\Delta F \approx 0.004\,\mathrm{kcal\,mol^{-1}\,\AA^{-1}}$. This value for the required reciprocal-space accuracy for dispersion interactions was chosen based on findings in Section 2.6.1 and is in good agreement with recent findings by Wennberg et al. [48] The resulting parameters are summarized in Table 2.5.

	β_1 [Å^{-1}]		h_1 [Å]		β_6 [Å^{-1}]	h_6 [Å]	
r_c [Å]	$i\mathbf{k}$	ad	$i\mathbf{k}$	ad	$i\mathbf{k}$ & ad	$i\mathbf{k}$	ad
7	0.373	0.375	1.3889	1.25	0.40	2.0	1.6667
8	0.328	0.324	1.5625	1.3889	0.33	2.5	2.0833
9	0.287	0.294	1.8519	1.3889	0.28	3.125	3.125
10	0.257	0.259	2.0833	1.6667	0.24	6.25	6.25

Table 2.5: Ewald parameter and grid spacing for PPPM validation simulations. The interpolation order in the simulations was $P = 5$. (Reprinted with permission from Ref. [81])

The parameters have been validated in a series of surface simulations of hexane modeled with the OPLS-AA force field [27] and SPC/E [72] water. If not otherwise explicitly stated, simulations were performed as described in Section 2.6.1. Results of the validation simulations are given in Tables 2.6 and 2.7. For each parameter combination, simulation results are within statistical uncertainties equal to results from Sections 2.7.2 and 2.7.3. Hence, inaccuracy introduced by PPPM parameters or the applied acceleration methods does not alter the simulation results and the performance results reported below were run with PPPM parameters that ensure accurate physics.

		$n_r = 1$		$n_r = 4$	
diff	r_c [Å]	ρ_{liq} [kg/L]	γ [mN/m]	ρ_{liq} [kg/L]	γ [mN/m]
ad	7	0.6427 (5)	16.40 (60)	0.6431 (8)	16.18 (60)
	8	0.6428 (7)	16.57 (70)	0.6425 (6)	16.52 (60)
	9	0.6428 (3)	16.22 (60)	0.6423 (3)	16.18 (60)
	10	0.6443 (7)	16.23 (50)	0.6437 (5)	16.12 (50)
$i\mathbf{k}$	7	0.6424 (7)	16.04 (50)	0.6428 (3)	16.27 (50)
	8	0.6432 (3)	16.30 (50)	0.6426 (6)	16.09 (50)
	9	0.6430 (8)	15.99 (60)	0.6423 (3)	16.18 (60)
	10	0.6446 (4)	16.08 (60)	0.6438 (6)	16.26 (50)

Table 2.6: Surface tensions and liquid densities of hexane when exploiting all of the methods to increase the efficiency mentioned in this section. (Modified reprint with permission from Ref. [81])

		$n_r = 1$		$n_r = 4$	
diff	r_c [Å]	ρ_{liq} [kg/L]	γ [mN/m]	ρ_{liq} [kg/L]	γ [mN/m]
ad	7	0.9962 (3)	60.28 (100)	0.9955(4)	60.46 (90)
	8	0.9958 (3)	60.94 (100)	0.9952 (2)	59.69 (90)
	9	0.9956 (3)	60.78 (90)	0.9952 (3)	60.38 (90)
	10	0.9954 (3)	60.05 (90)	0.9962 (2)	58.41 (90)
$i\mathbf{k}$	7	0.9970 (3)	59.60 (90)	0.9962 (3)	59.49 (90)
	8	0.9964 (3)	59.82 (90)	0.9956 (3)	60.68 (80)
	9	0.9961 (4)	60.15 (80)	0.9955 (4)	60.20 (90)
	10	0.9964 (4)	57.96 (80)	0.9956 (5)	58.71 (90)

Table 2.7: Surface tensions and liquid densities of SPC/E water when exploiting all of the methods to increase the efficiency mentioned in this section. (Modified reprint with permission from Ref. [81])

2.8.2 Performance simulations

We have performed a series of performance runs on the Vesta Blue Gene/Q system at the Argonne Leadership Computing Facility. To show that our findings are transferable to other architecture types, we performed similar performance tests on the SuperMUC Sandy Bridge-EP processors at the Leibniz Rechenzentrum in Garching, Germany. Performance simulations have been performed with the 22Mar13 version of LAMMPS. The parallelization strategies for the parts of the code used here can be found in [84,85].

2.8.2.1 Simulation setup

We have chosen SPC/E water as the test case for our performance comparisons. The model is widely known and used in the molecular simulation community. Furthermore, it requires long-range Coulomb solvers, while also allowing use of long-range dispersion solvers. As discussed in Section 2.3, the performance of the latter is dependent on the chosen mixing rule. To examine this effect simulations were run with routines that compute long-range dispersion interactions for the popular combinations of the Berthelot [86] mixing rule for energies

$$\epsilon_{ij} = \sqrt{\epsilon_{ii}\epsilon_{jj}}, \tag{2.79}$$

with the Lorentz [87] and Good-Hope [88] mixing rules for the LJ σ-parameters:

$$\sigma_{ij} = \tfrac{\sigma_{ii}+\sigma_{jj}}{2}, \quad \text{(Lorentz)} \tag{2.80}$$

$$\sigma_{ij} = \sqrt{\sigma_{ii}\sigma_{jj}}. \quad \text{(Good-Hope)} \tag{2.81}$$

Combinations of the Good-Hope and the Berthelot (GHB) rule are computationally favorable because the dispersion coefficients C_{ij} follow a geometric mixing rule. Performance results reported below for this mixing rule are representative for all mixing rules that provide geometric mixing of the dispersion coefficients [89, 90].

To create an initial starting configuration, a small system of 666 SPC/E molecules has been equilibrated in a cubic box with a volume of 20191 Å^3, which corresponds to a density of approximately 1 kg/L. This system has been equilibrated for 20 ps in the NVT ensemble at 300 K. Configurations for the performance simulations were obtained by simply replicating the equilibrated system. Performance simulations were run for 1 ps.

We have performed a series of simulations with the PPPM method for dispersion and with truncated potentials. Simulations using truncated potentials were performed with cutoff radii of 10 and 12 Å as these are typically used values for bulk phase simulations. Simulations with the PPPM dispersion method were performed with cutoff radii of 7, 8, 9, and 10 Å. As shown in Section 2.8.1.3, simulation results are independent of cutoffs, provided the remaining parameters are chosen appropriately. These cutoffs can therefore be considered as tuning parameters. All comparisons are between a simulation using the PPPM method for dispersion to a simulation using a truncated potential, with all other features and settings kept the same.

We performed weak-scaling and strong-scaling tests. In the weak-scaling tests, the average number of particles per task (where the number of tasks is the number of MPI tasks times the number of OpenMP threads per MPI task) was kept constant at 999.

On the BG/Q machine, the number of tasks in the weak-scaling test were 128, 1024, 8192, and 65536, corresponding to 127 872, 1 022 976, 8 183 808, and 65 470 464 particles. In strong-scaling tests, the overall number of particles was kept constant at

127 872. The number of tasks were 128, 256, 512, 1024, and 2048. We also examined the effect of maximizing the performance with hybrid parallelization. We have varied the number of OpenMP threads per MPI task between 2, 4, and 8 for all simulations. Additionally, we have performed all simulations with and without rRESPA. rRESPA settings are as reported in Section 2.8.1.3.

Performance runs on SuperMUC were performed the same way as those on Vesta, except that the number of OpenMP threads per MPI rank were 1, 2, and 4 in simulations on SuperMUC. Furthermore, weak-scaling tests were performed with 16, 128, 1 024, and 8 192 tasks. Strong scaling tests were performed with 127 872 particles with 64, 128, 256, 512, 1 024, and 2 048 tasks; and with 15 984 particles with 16, 32, 64, 128, and 256 tasks.

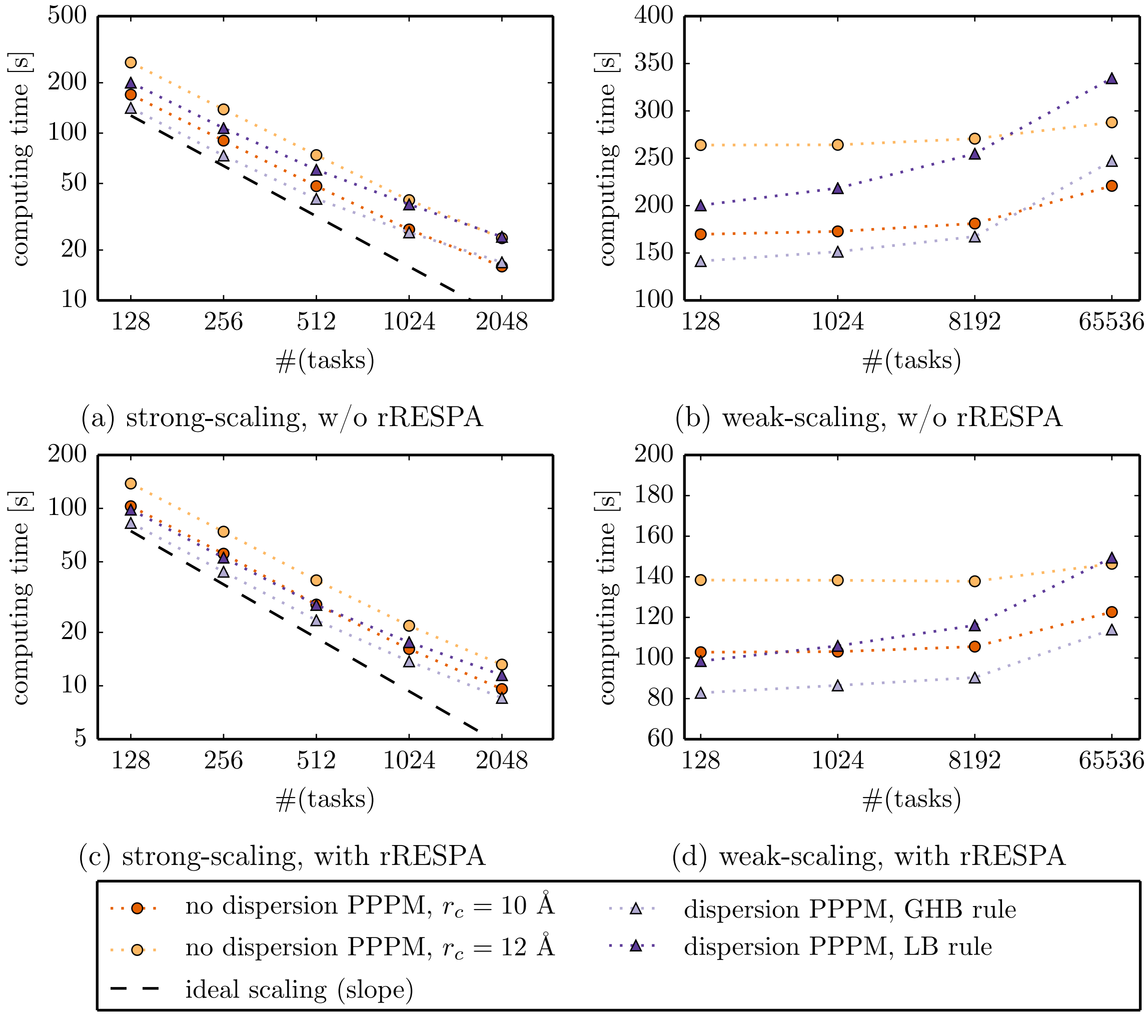
The PPPM parameters for Coulomb interactions were selected as described in Section 2.8.1.3. Parameters for the PPPM method for dispersion interactions are almost identical to those in Table 2.5. Slightly different values for the grid spacing were chosen because the number of grid points in each direction is restricted to values that can be factored by 2, 3, and 5. Analytic differentiation has been used in the performance simulations. Evaluation of the pair potentials was accelerated using tabulation.

2.8.2.2 Results

In this section, we report simulation times for settings that provided maximum performance for a given system and number of tasks. For simulations run without PPPM for dispersion, results for $r_c = 10\,\text{Å}$ and $r_c = 12\,\text{Å}$ are reported because these are the most frequently used cutoffs for dispersion terms in popular force fields. For simulations with PPPM for dispersion, the cutoff is a tuning parameter and we therefore report only the values that provide the fastest computations. Figures 2.20 and 2.21 show the total simulation time of the performance runs as a function of the number of tasks for the performance runs on Vesta and SuperMUC.

In simulations without rRESPA and with a fixed number of particles (cf. Figure 2.20a), fastest simulations were obtained when the PPPM method for dispersion was used with the GHB mixing rule. Shorter simulation times were only obtained without the long-range dispersion solver for systems with around 60 particles per task. Simulations with PPPM for dispersion and the LB mixing rule usually outperformed simulations without the PPPM method for dispersion and a cutoff of $12\,\text{Å}$. Similar behavior is observed in tests with an increasing number of particles, as shown in Figure 2.20b.

In simulations with rRESPA, the best performance in all test cases was achieved



(a) strong-scaling, w/o rRESPA

(b) weak-scaling, w/o rRESPA

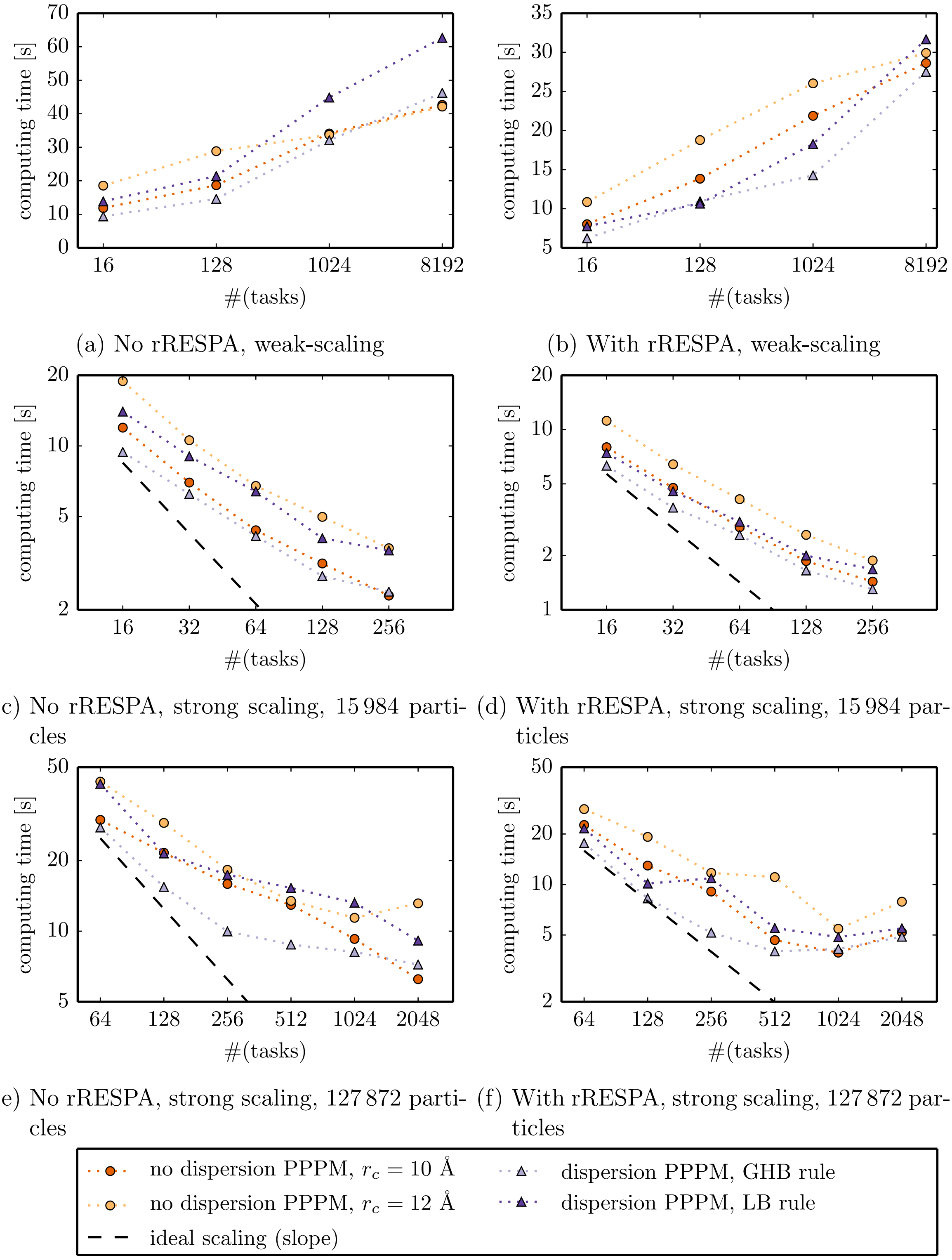
(c) strong-scaling, with rRESPA

(d) weak-scaling, with rRESPA

Figure 2.20: Performance of simulations that provided fastest computations for a given number of particles and CPUs on Vesta. (Modified reprint with permission from Ref. [81])

when using the PPPM method for dispersion with the GHB mixing rule. For the lowest number of particles or the highest load of particles per core, the speed-up compared to simulations with a truncated potential and a cutoff of 12 Å is more than 40 %. Even the unfavorable LB mixing rule often provided a better performance than simulations with a truncated potential cutoff of 10 Å. The PPPM method for dispersion with the LB mixing rule was outperformed by the simulations using a truncated potential and a cutoff of 12 Å only in the simulations of the largest system.

When moving towards larger numbers of particles or a larger number of processors, larger cutoffs become more favorable, and the PPPM method for dispersion becomes computationally less advantageous. These observations are related to the limitations of parallel FFTs, which require data to be contiguous in one dimension and thus can be



(a) No rRESPA, weak-scaling

(b) With rRESPA, weak-scaling

(c) No rRESPA, strong scaling, 15 984 particles

(d) With rRESPA, strong scaling, 15 984 particles

(e) No rRESPA, strong scaling, 127 872 particles

(f) With rRESPA, strong scaling, 127 872 particles

Figure 2.21: Performance of simulations that provided fastest computations for a given number of particles and CPUs on the Sandy-Bridge architecture at SuperMUC. (Modified reprint with permission from Ref. [81])

parallelized only in two dimensions. They also require a redistribution of data before, in between, and after individual parallelized transforms, leading to increased communication with more processors. For very large problems and a very large number of processors it is faster to do the reciprocal-space calculation with a subset of communicating processors via hybrid MPI–OpenMP parallelization or on a separate partition of processors. At the very extreme scale a multigrid approach offers better scaling behavior in exchange for a larger prefactor, but uses the same kind of short- versus long-range split in the calculation of the interactions [32, 38]. We anticipate a similar performance benefit by including long-range dispersion interactions for these solvers with reduced short-ranged cutoffs, too. As PPPM for dispersion requires additional FFTs and the number of grid points for the Coulomb interactions is larger when using smaller cutoffs, the benefits and shortcomings of the FFTs dominate for simulations with the long-range dispersion solver; exploring options to reduce the communication overhead for these cases is thus of importance.

Trends of the simulation results on SuperMUC are similar to those that are observed on Vesta. Important differences are (i) that the scaling behavior is better on Vesta and (ii) that simulation results on SuperMUC are subject to stronger fluctuations. For example, results in Figure 2.21f without PPPM for dispersion and a cutoff of $r_c = 12\,\text{Å}$ with 512 tasks or with PPPM for dispersion and the LB mixing rule with 256 tasks deviate from the general trend of the lines. Moreover, results in Figure 2.21b for 128 tasks, where the simulations with the LB rule are more efficient than simulations with the GHB rule, can only be caused by noise in the results. These uncertainties in the simulation results are almost certainly caused by the node selection policies on the SuperMUC, which did not guarantee the same network configuration for different jobs in our trials.[1]

2.8.3 Discussion

Whether the PPPM method for dispersion can be used to decrease the simulation time depends on multiple factors, including the system size, the level of parallelization, the employed mixing rule, the machine used for the simulations, as well as what cutoff would be considered appropriate in simulations using a truncated potential. Moreover, the use of rRESPA has an even stronger influence on the results. Therefore drawing universally valid conclusions is difficult, yet it is possible to state that a performance

[1] In principle it is possible to obtain the same network configuration when this is specified in the input scripts. According to the recommendations of the LRZ, we decided against that to shorten the time that our simulations were pending.

gain using a PPPM method for dispersion was achieved in most of the test cases described here. This includes cases using the computationally unfavorable LB mixing rule. Moreover, even when using a long-range dispersion solver did not lead to a better performance, the disadvantage was not unaffordably large.

When comparing performance, we note that simulations with the PPPM method for dispersion provide better accuracy than simulations without a long-range solver. Especially for strongly inhomogeneous systems, much larger cutoffs (e.g., $r_c = 23\,\text{Å}$ as suggested in [26]) are required to provide accurate results. Such large cutoffs are much more expensive than any of the results reported in this study. Hence, the PPPM method for dispersion substantially increases the accuracy with gains or, at worst, minor losses in efficiency, depending upon the simulation settings. Its application can therefore be advantageous even when the physics of a system do not seem to demand the use of a long-range dispersion solver. We note that our approach is also applicable to other dispersion potentials, such as force fields that use an r^{-7} term. However, for force fields like the Born potential [91, 92] that contain multiple dispersion terms (such as linear combinations of r^{-6}, r^{-8}, and r^{-10}), the technique will become less advantageous because each term requires a separate reciprocal-space computation.

For a given simulation, one must consider not only how including the long-range dispersion solvers affects the efficiency, but also how the physical behavior of simulations changes and whether this change is desirable. The physical nature of dispersion is not a simple decay to zero beyond the cutoff. However, many force fields truncate dispersion interactions for computational reasons. The parameters in those force fields are thus partly designed to compensate for errors caused by truncation of the dispersion potential at the cutoff. Applying a long-range dispersion solver thus usually means that the force field is used differently than intended.

Whether this alternate usage is acceptable or even desired depends on the simulation and the force field. As described in Section 2.1, the need for long-range dispersion interactions is generally accepted in strongly inhomogeneous simulations, which is why the deviation from the parameterization settings of the force field is of interest here. For force fields whose dispersion coefficients were not determined from fitting but, for example, from *ab initio* calculations, using long-range dispersion solvers is also beneficial. For force fields that have been parameterized either using a large cutoff (such as GROMOS96 [80] with a cutoff of 14Å) or small cutoffs with long-range corrections, such as OPLS-AA [27] or TraPPE [79], the difference in obtained physical results might be sufficiently small that the decision whether to use a PPPM method for dispersion can be made solely by focusing on the performance. These considerations are in agreement with recent results by Wennberg et al. [48]. For force fields designed to be used with small cutoffs and without long-range corrections, the differences in simulation results are expected to be relatively large. A long-range solver may not be advantageous under such conditions.

Optimal results will be achieved, however, when the solver is used together with a force field parameterized taking long-range dispersion interactions into account. As the use of PPPM for dispersion facilitates simulations that are more accurate and more efficient, we suggest that future force-field development should take long-range dispersion interactions into account alongside long-range Coulomb interactions.

2.9 Interlacing vs. non-interlacing for dispersion in practice

It has been shown in several articles that interlacing can provide higher accuracy for given PPPM parameters and therefor has the potential to provide more efficient simulations for a desired accuracy [63,69,93]. This potential gain is also suggested by our results in Section 2.5. This finding is true, however, only for a certain parameter and accuracy domain. Whether a speedup can be observed in practical applications will thus depend on whether parameters providing optimal performance at the required accuracy are in the parameter domain where the interlaced algorithm is superior. In this section we present a parametric study with an atomistic LJ system in which we address this issue.

The simulation setup is similar to that in Section 2.7.1, with the difference that we examined different sets of parameters for PPPM for dispersion. We use interpolation order $P = 5$, ad differentiation, use cutoffs of $r_c \in \{2.5, 3.0, 3.5, 4.0\}\,\sigma$ with corresponding Ewald parameters of $\beta_6 \in \{1.18, 0.9, 0.7, 0.56\}\,\sigma^{-1}$ to ensure that all simulations are run at approximately the real-space accuracy of $\Delta F_{\mathrm{real}} = 0.001\epsilon/\sigma$ that was found to provide accurate physical results, and with grid points in x direction $n_x \in \{2, 4, 8, 12, 16\}$ whereat the grid spacing was equal in each Cartesian dimension and was approximately $h \in \{5.5, 2.75, 1.38, 0.92, 0.69\}\,\sigma$. The chosen temperature is $0.8\,\epsilon/k_B$. Surface tensions and liquid densities were determined as described in Appendix A, with instantaneous surface tensions computed every timestep and particle coordinates stored every 1000 timesteps to compute liquid densities. The force error was measured by comparison of a single well-equilibrated snapshot of the simulation to results obtained with very accurate computations. Single-core timings were obtained similar to Section 2.5.3 with the difference that the computations were performed on a single core of a Intel Westmere processor @ 3.07GHz.

The physical results, force errors, and the timing data from our simulations are shown in Table 2.8. With decreasing force error, surface tensions and liquid densities converge to $\rho_{liq} \approx 0.796 \pm 0.0005\sigma^{-3}$ and $\gamma \approx 0.93 \pm 0.01\epsilon\sigma^{-2}$. Simulations run with sufficient accuracy to provide results in agreement with this high-accuracy limit are printed with bold font. Both the interlaced and non-interlaced algorithm converge to these limits with increasing accuracy, which shows that both algorithms provide accurate results when parameters are selected appropriately.

The maximum force error below which the high accuracy limit is reached in the simulated data varies between the different cutoffs an PPPM variants. For example, we observe a highly inaccurate simulated surface tension for the normal PPPM with $r_c = 4\,\sigma$ and $h = 5.5\sigma^{-1}$ with a reciprocal space accuracy of $0.0263\,\epsilon/\sigma$ whereas accurate results are observed for the interlaced version with $r_c = 3.5\,\sigma$ and $h = 2.75\sigma^{-1}$ and a higher reciprocal-space error of $0.0504\,\epsilon/\sigma$. For reciprocal-space force errors below $0.01\,\epsilon/\sigma$, results have converged to the high-accuracy limit for all simulations.

The overall best performance at sufficient accuracy to obtain accurate physical results was obtained with the non-interlaced version of the algorithm with $r_c = 3\,\sigma$ and $h = 1.38\sigma^{-1}$ with a computation time t_{step} of approximately 12 ms per timestep. The best performance with the interlaced version of the algorithm that provided correct physical results of the system was obtained with $r_c = 3\,\sigma$ and $h = 1.38\,\sigma^{-1}$ with a computation time of approximately 18 ms per timestep, or approximately 50 % slower. The non-interlaced version of the algorithm thus strongly outperforms the interlaced version for the test case reported here. It should be noted that the non-interlaced version of the algorithm is also superior to the interlaced version when comparing each selected cutoff separately. Similar tests were performed with interpolation orders $P = 3$ and $P = 4$ and the $i\mathbf{k}$ differentiation. Results with these settings showed the same trends as those reported here.

Before concluding this section we briefly comment on the choice of optimal simulation parameters that provide maximum efficiency while providing accurate results. When, for example, randomly selecting the cutoff value as $r_c = 2.5\,\sigma$, the best achieved performance is approximately a factor of two smaller than what can be obtained with the improved choice for the cutoff of $r_c = 3.0\,\sigma$. Tools that can generate optimal parameters to avoid pitfalls like the one just mentioned can thus be highly beneficial and are eagerly desired.

r_c [σ]	solver	h [σ]	ρ_{liq} [σ^{-3}]	γ [ϵ/σ^2]	$\Delta F_{\text{reciprocal}}$ [ϵ/σ]	t_{step} [ms]
2.5	normal	5.5	0.8965(397)	15.076(30)	1.73×10^{0}	9.66(5)
		2.75	0.8519(534)	7.751(26)	1.64×10^{0}	9.82(9)
		1.38	0.8099(80)	1.823(26)	8.36×10^{-1}	11.17(15)
		0.92	0.8044(38)	1.510(27)	2.69×10^{-1}	14.35(9)
		0.69	0.7980(6)	0.945(29)	7.53×10^{-2}	20.00(10)
	interlaced	5.5	0.8119(12)	0.680(27)	1.72×10^{0}	13.49(20)
		2.75	0.8113(4)	0.874(27)	1.55×10^{0}	13.75(5)
		1.38	0.8041(4)	1.014(25)	6.16×10^{-1}	16.08(16)
		0.92	0.7992(3)	0.976(25)	1.12×10^{-1}	16.03(7)
		0.69	**0.7972(8)**	**0.947(25)**	**5.90×10^{-3}**	**30.58(24)**
3.0	normal	5.5	0.8415(488)	6.858(30)	4.03×10^{-1}	11.97(9)
		2.75	0.8132(81)	3.010(27)	3.30×10^{-1}	11.97(5)
		1.38	**0.7969(4)**	**0.941(28)**	**9.92×10^{-2}**	**12.01(8)**
		0.92	**0.7934(68)**	**0.950(29)**	**1.91×10^{-2}**	**14.08(10)**
		0.69	**0.7966(13)**	**0.918(26)**	**3.71×10^{-3}**	**19.74(17)**
	interlaced	5.5	0.7988(7)	0.746(27)	3.95×10^{-1}	15.50(20)
		2.75	0.7993(6)	0.878(28)	2.86×10^{-1}	15.96(8)
		1.38	**0.7971(12)**	**0.933(29)**	**5.14×10^{-2}**	**18.32(5)**
		0.92	**0.7966(7)**	**0.959(29)**	**1.82×10^{-3}**	**18.38(7)**
		0.69	**0.7963(2)**	**0.946(26)**	**2.96×10^{-4}**	**32.90(33)**
3.5	normal	5.5	0.8072(316)	15.689(27)	1.02×10^{-1}	14.97(6)
		2.75	0.8007(39)	1.383(26)	6.83×10^{-2}	15.09(11)
		1.38	**0.7966(11)**	**0.917(27)**	**1.04×10^{-2}**	**16.59(15)**
		0.92	**0.7966(6)**	**0.944(26)**	**1.24×10^{-3}**	**19.80(10)**
		0.69	**0.7960(7)**	**0.943(26)**	**2.26×10^{-4}**	**25.43(19)**
	interlaced	5.5	0.7961(8)	0.823(25)	9.66×10^{-2}	18.78(22)
		2.75	**0.7971(10)**	**0.919(26)**	**5.04×10^{-2}**	**19.17(14)**
		1.38	**0.7961(11)**	**0.934(25)**	**2.62×10^{-3}**	**21.44(8)**
		0.92	**0.7964(5)**	**0.951(26)**	**8.96×10^{-5}**	**21.51(34)**
		0.69	**0.7964(8)**	**0.948(25)**	**9.72×10^{-6}**	**36.08(51)**
4.0	normal	5.5	0.8280(577)	5.515(27)	2.63×10^{-2}	18.95(11)
		2.75	**0.7960(9)**	**0.929(27)**	**1.30×10^{-2}**	**19.00(10)**
		1.38	**0.7956(9)**	**0.936(26)**	**8.90×10^{-4}**	**20.51(18)**
		0.92	**0.7958(16)**	**0.936(26)**	**8.27×10^{-5}**	**23.63(12)**
		0.69	**0.7962(6)**	**0.904(24)**	**1.77×10^{-5}**	**29.46(21)**
	interlaced	5.5	0.7958(8)	0.875(26)	2.34×10^{-2}	22.65(19)
		2.75	**0.7957(3)**	**0.911(24)**	**7.23×10^{-3}**	**22.92(7)**
		1.38	**0.7955(5)**	**0.930(26)**	**2.69×10^{-5}**	**25.30(15)**
		0.92	**0.7953(6)**	**0.918(27)**	**4.07×10^{-6}**	**25.31(11)**
		0.69	**0.7957(10)**	**0.933(26)**	**7.44×10^{-7}**	**40.06(46)**

Table 2.8: Liquid densities, surface tensions, reciprocal-space errors and single-core timings for the LJ system. Simulations that provide corrected physical results are printed bold. Among those, the fastest computations were obtained with non-interlaced PPPM.

2.10 Summary

In this chapter we present the development of a PPPM algorithm for computing long-ranged dispersion interactions, which included a description of the algorithm, error estimates, and a couple of numerical tests that highlight the features of the algorithm and the error estimates. We furthermore perform a series of tests in which the utilization of the PPPM method in practice is examined.

An important finding confirmed by numerical studies and theoretical considerations is that for the dispersion PPPM algorithm, the real-space error has a stronger impact on physical results than the reciprocal-space error. Parameters for the algorithm must thus be chosen such that the real-space error is considerably smaller than the reciprocal-space error to obtain simulations that are both accurate and efficient. This is the major difference to PPPM for Coulomb interactions, where the choice of parameters is optimal when parameters are selected such that the real- and reciprocal-space errors are equal.

The effect of including long-range dispersion interactions on physical results versus simply truncating dispersion interactions was examined for a model LJ system as well as for the realistic systems of water, in which Coulomb interactions dominate, and hexane, in which dispersion interactions dominate. In systems in which Coulomb interactions dominate, the effect on the physical behavior of truncating the dispersion potential was rather weak. The inclusion of long-range dispersion interactions for these systems is important either when very accurate physical results are required, or when *a posteriori* corrections are not applicable. For systems in which dispersion interactions dominate, truncation of the pair potentials has strong effects on the results. The inclusion of long-range dispersion solvers is beneficial for these systems even when only moderate accuracy is required.

Because the effect of the cutoff on the physical results is eliminated when using PPPM for dispersion, as long as the cutoff is not chosen so small that truncation of the repulsive potentials starts to influence the results, the cutoff is no longer a parameter to tune the accuracy of the simulations and influence the physical results, but a parameter to tune the efficiency. It was shown that exploiting this feature of PPPM for dispersion can lead to simulations that are not only more accurate but also more efficient than those that use a simple truncation of dispersion interactions. This finding disproves the frequently raised argument against PPPM for dispersion solver that the gain in accuracy is unaffordable because of the increased computational cost. Because simulations using PPPM for dispersion are not only more accurate but also faster than simulations in which dispersion is simply truncated, incorporation of long-range dispersion interactions is also beneficial in simulations where it is not required from the physics of the system.

Finally, we test the interlaced version of the algorithm. We confirm that this modification can increase the efficiency for high accuracy. For the test case considered here, however, using the interlaced algorithm was not beneficial in practical applications because moderate accuracy, at which the non-interlaced PPPM method is faster, was sufficient to obtain accurate physical results.

3 The molecular mechanism of superspreading

3.1 Literature review and motivation

Superspreading is the ultra-rapid wetting of aqueous droplets on hydrophobic substrates facilitated by trisiloxane surfactants. The phenomenon is characterized by ultra-fast wetting dynamics and final contact angles that are too small to be measured. The effect depends on the substrate hydrophobicity, the chain lengths of the hydrophilic tail, and the surfactant concentration. Superspreading is maximized for intermediate values of all of these factors and vanishes if any of these factors deviate too far from its optimum. Of particular interest for the surfactant concentration is that maximal spreading rates are achieved just below a bulk-phase transition from surfactant vesicles to lamellar phases. Because of its relevance in fundamental physics and technical applications, there have been numerous studies on this phenomenon in the last 20 years. Yet, it is still not understood how or why trisiloxane surfactants facilitate superspreading [3, 10, 18, 94–98].

Spreading dynamics of droplets can often be described by a power law,

$$r \propto t^{\alpha}, \tag{3.1}$$

where r is the base radius of the droplet, t is the time, and α is the spreading exponent characterizing the spreading regime. The exponent α results from a balance of driving and refraining forces and depends on the underlying physics [10]. For small droplets, the driving force is typically differences in surface free energies, whereas the refraining force can vary. After depositing a droplet on a substrate, the spreading passes through different regimes. Right after deposition, the refraining force is inertia; the spreading exponent in this regime is $1/5 \leq \alpha \leq 1/2$ [99]. After some characteristic time that depends on the droplet size and fluid properties, the spreading decelerates. The refraining force in this regime is viscous dissipation at the contact line. Lubrication theory and the no-slip boundary condition between the substrate and the droplet predict the Huh–Scriven paradox of logarithmically diverging viscous dissipation at wedge-like shaped contact lines [10,12]. Obviously, the approximations have to break down at the contact line because such a result is in conflict with the observed spreading of droplets. However, this finding illustrates why the region in the vicinity of the contact line is important for the viscous spreading regime.

Various models have been proposed to overcome the singularity at the contact line, the most commonly used of which are the hydrodynamic model (HDM) and molecular kinetic theory (MKT). In the HDM, singularities at the contact line are avoided by allowing slip [10, 11]. In MKT, contact-line motion is described via molecular jumps [100]. The models predict different spreading exponents: HDM and MKT predict $\alpha_{\mathrm{3D,HDM}} = 1/10$ and $\alpha_{\mathrm{3D,MKT}} = 1/7$ in three dimensions, and $\alpha_{\mathrm{2D,HDM}} = 1/7$ and $\alpha_{\mathrm{2D,MKT}} = 1/5$ in two dimensions. The 3D spreading exponent for the HDM corresponds to the most frequently observed spreading exponent and matches "Tanner's law" [10,17].

Recently, Wang et al. [101] showed that aqueous solutions of trisiloxane surfactants on hydrophobic substrates pass through the regimes described above. They observed two major differences between superspreading and non-superspreading solutions of trisiloxane surfactants. First, in the viscous spreading regime, the spreading exponent of the non-superspreading surfactants agreed with Tanner's law for $\alpha \approx 1/10$. For the superspreading surfactant, the spreading exponent was slightly larger, with a value of $\alpha \approx 1/7$. The authors concluded that the HDM describes non-superspreading wetting well, while the underlying mechanism for superspreading is closer to the MKT. Second, while spreading remained in the viscous regime for the non-superspreading surfactant, the superspreading droplet entered a third spreading regime, the superspreading regime. In their study, spreading exponents in this regime were $0.25 \lessapprox \alpha \lessapprox 0.5$. Other authors have measured similar or larger spreading exponents for this regime [102].

Numerous hypotheses have been published attempting to explain the superspreading mechanism [102–108]. Apart from the theory of Kabalnov [104], which attributes superspreading to large negative spreading coefficients facilitated by vesicle formation, the contact-line region of the droplet is commonly considered crucial for superspreading. Specifically, hypotheses or models that explain superspreading are based on various assumptions: for instance, that vesicles unzip at the contact line [103], that a thick precursor stabilized by vesicles is formed [105], that surfactants diffuse rapidly to the three phase contact line from the bulk [102], that surfactants adsorb directly from the liquid–vapor interface to the solid–liquid interface [107], or that surfactants form a bilayer that precedes the droplet [108].

For most of these theories, it is unclear how exactly the hindrance of rapid spreading because of extreme viscous dissipation at the contact line should be resolved. However, Karapetsas et al. [107] showed in computational fluid dynamics simulations that the extreme dissipation can be overcome and superspreading can occur if Marangoni stresses arise and are maintained during spreading. In their study Marangoni stresses and surfactant concentrations gradients resulted from surfactants being both soluble and being able to adsorb directly through the contact line from the liquid surface to the solid–liquid interface. The solubility of trisiloxane surfactants is well-known, but it is unclear how the direct adsorption through the contact line works; formation of a bilayer ahead of the droplet was hypothesized. The study proves that superspreading can be facilitated by Marangoni stresses but does not provide evidence for how these stresses develop and are maintained. The underlying molecular mechanisms remain subject to speculation.

As suggested by Maldarelli [109], MD simulations are required to further understand the superspreading phenomenon. Indeed, there have already been several MD simulations related to surfactant enhanced spreading and superspreading in particular. As part of a larger study Nikolov et al. [110] presented simulations with a trisiloxane surfactant at the air–water interface. They find that the head group of the trisiloxane has a very compact shape and related this to superspreading. Visualization of their results, however, suggests that the hydrophilic tail of their surfactant was stretched, whereas it is known from experiment that the tail has a helical configuration in water [111], indicating that the employed model is inaccurate. McNamara et al. [112] performed simulations with simple model molecules, finding that linear surfactants can enhance wetting, especially when there are strong interactions between the substrate and the hydrophobic part of the surfactant. This finding is also confirmed in a study by Kim et al. [113]. In a simulation study with stronger connection to superspreading by Shen et al. [114], the influence of the shape of the surfactant, such as linear or T-shaped (as in trisiloxanes), has been compared using model LJ surfactants. It was found that T-shaped surfactants lead to faster spreading rates and smaller final contact angles. Moreover, it was found that under certain conditions T-shaped surfactants can promote bilayer formation. Finally, Halverson et al. [115] have performed MD simulations of spreading droplets with more complex, realistic models. While simulations with alkyl ethoxylate surfactants showed behavior in agreement with experiment, the trisiloxane-laden droplet did not spread. It later turned out that realistic behavior of the simulations was hindered by the trisiloxane model, which failed to reproduce adequately the strong reduction in the surface tension of water caused by trisiloxane surfactants [116].

While the application of MD to directly observe the superspreading effect is not promising, because the effect possibly involves macroscopic phenomena, i.e., surfactant concentration gradients at the interfaces and flows driven by the resulting Marangoni stresses, MD has the potential to observe the molecular mechanisms that underlie the mechanism. The identification of these mechanisms is the primary target of the work described in this chapter.

To better understand superspreading using MD simulations, more realistic and reliable surfactant force fields are required. We first present an atomistic model suitable for studying superspreading. This model contains parameters to describe alkane, perfluoroalkane, dimethylsiloxane, and poly(ethylene oxide) chains and their interactions with water. It thus contains all building blocks that are required to model surfactant laden water droplets and polymeric substrates. The model parameters are modifications of existing quantum chemistry-based models of the aforementioned polymers [53,54,117,118]. These models have been optimized for use with the TIP4P/2005 water model [119], which is especially suitable for use in interfacial systems because of its accurate representation of interfacial properties, such as surface tension [120]. The model is then used in simulations of surfactants at the water interface to validate the model and assess its potential application to spreading simulations.

Finally, we perform large-scale MD simulations of water droplets laden with different trisiloxane and alkyl ethoxylate surfactants on various substrates. We cover different conditions that correspond to superspreading and non-superspreading scenarios in experiment. Specifically we cover substrates from too hydrophilic to too hydrophobic and surfactants from too short to too long, an intermediate state in which superspreading occurs in experiment, and a reference simulation with alkyl ethoxylate surfactants. We report a smooth contact-line transition at superspreading conditions similar to that observed by McNamara et al. [112]. The relation of this mechanism to the Huh–Scriven paradox [12] is briefly addressed and the relevance to superspreading is discussed. The observed mechanism complements the simulation study by Karapetsas et al. [107] and allows plausible explanations for the complex behavior of the superspreading mechanism on the surface energy of the substrate and the the size of the surfactants.

The remainder of this chapter is structured as follows. Section 3.2 covers the force field development and validation with model compounds and surfactants at the water interfaces. The spreading simulations are described in Section 3.3. We offer concluding remarks in Section 3.4.

3.2 Molecular modeling of surfactant mixtures

3.2.1 Modeling depth and initial potentials

There are several requirements for a good force field for a given application. First, all relevant physical properties should be reproduced with sufficient accuracy. For the problem considered here, this includes the surface tensions of the simulated materials, as surface tension is the driving force in droplet spreading. For superspreading it is unclear which additional properties are most influential. As described in the Section 3.1, the phase behavior of the surfactant in aqueous solutions might play a dominant role. Additionally, the influence of the surfactants on the bending rigidity of the droplet could influence the spreading. Moreover, more subtle features, such as the extraordinary flexibility of the Si-O-Si angle, might play an important role and should be captured well. In addition, a good force field should be consistent, which means that simply combining existing models can lead to inaccurate results, as observed in [121]. Finally, the model must be computationally simple enough to allow its application to the intended problem.

The systems relevant in this study typically require a moderately large number of particles. Even the interfacial simulations presented in Section 3.2.6 already consist of around 30 000 to 40 000 atoms, and the simulations of spreading droplets in Section 3.3 have more than 300 000 atoms. Computationally costly models, like ab initio models, are thus infeasible. Polarizable models [122,123] are computationally less intensive, but still demanding. For water surfaces that do not contain ions or highly polar moieties, or are close to anisotropic environments, such as those considered here, the gain in accuracy by using polarizable models compared to nonpolarizable models is small [123–126]. Because the expected gain in accuracy is small and the model should be applicable in simulations with a large number of particles, we use nonpolarizable models with a slight loss of accuracy in favor of the massive gain in computational efficiency.

At the other end of the complexity scale are coarse-grained (CG) models. At that level, there has already been very successful work in the development of force fields for surfactant–water interactions [127–131]. However, we have several concerns about applying CG models to the problem considered here. First, CG models are usually dedicated for a special purpose, so properties not targeted in the development process are usually not reproduced well. Since neither the properties which enable superspreading, nor the mechanism itself are clear, starting with a CG model seems unwise. Moreover, CG models do not provide atomistic detail, which complicates the simultaneous accurate representation of multiple features, such as the flexibility of the Si-O-Si angle and the bulkiness of the trisiloxane head group. Finally, the starting point for the development of CG models are typically atomistic models. Thus, reliable atomistic models must be available before we can construct CG models [132].

From the considerations above, a non-polarizable atomistic model appears the proper starting point. Before proceeding to the surfactant model, we first address the choice of water model. For spreading, an important criterion is a model's ability to reproduce the high surface tension of water. Unfortunately, most popular, non-polarizable water models such as TIP3P, TIP4P [133], and SPC/E [72], consistently underestimate the surface tension. For example, deviations from experiment are more than 10% for SPC/E and almost 30% for TIP3P at 300 K [120]. The only currently available generic model whose surface tension agrees well with experiment is the TIP4P/2005 model [119], which can also reproduce well a number of other properties [120]. Hence, we select the TIP4P/2005 water model for our simulations.

To build the trisiloxane surfactant molecule we follow the typical approach of assembling it from molecular building blocks. The most popular generic models from which such building blocks can be obtained have been designed for biomolecular simulations, such as OPLS [27], GROMOS [134], CHARMM [135], and AMBER [136], or phase equilibria, such as TraPPE [79]. While the former are typically optimized for use with a given water model, the latter typically do not contain water. Since the dedicated water model cannot easily be replaced by TIP4P/2005, such force fields cannot be used for our simulations without intensive testing. Another problem related to these force fields is that they typically do not contain parameters to describe dimethylsiloxanes, which is a necessity for superspreading. In principle, one could combine a generic model with a specialized dimethylsiloxane model, such as those of Sun et al. [137] or Frischknecht et al. [138], but this is the approach already tried by Halverson et al. [115] without success. Moreover, these specialized models also might not work well with the TIP4P/2005 model.

The models that meet our requirements best are quantum chemistry-based models for polyalkene [53], poly(ethylene oxide) (PEO) [117], polytetrafluoroethylene (PTFE) [54], and polydimethylsiloxane (PDMS) [118]. Although these force fields are seemingly independent, the interactions of these force fields have been determined using similar quantum-chemistry methods and have not been determined from empirical fits, indicating that they should be transferable. Furthermore, the alkane force field of [53] was used to generate the force fields in [54,117,118]. Finally, the potentials and parameters are consistent throughout these force fields: for instance, the parameters for the van der Waals (vdW) interactions for C, H, and O atoms are the same across all the force fields.

To fully satisfy our needs, we made three modifications to these models. First, our developed force field is intended for interfacial simulations in which the application of long-range dispersion solvers is required to obtain accurate physical behavior, as discussed in Chapter 2. Because maximum efficiency in simulations with long-range solvers is achieved if dispersion coefficients follow a geometric mixing rule, coefficients of the original force field were reparameterized such that this condition is met. Theoretical arguments justifying our modifications are presented in Section 3.2.3.1. Next, the model was extended to work well with the TIP4P/2005 water model. Finally, for surfactants and copolymers composed of different parts of these polymers, a few bonded interactions close to the linking points not available in the literature were parameterized.

3.2.2 Functional form

The functional form of the force field is only briefly presented here. In particular, we describe the treatment of vdW interactions, because these were the parameters requiring most modifications. The original force field for the polymers [53,54,117,118] that we are modifying describes vdW interactions with a Buckingham potential

$$U_{\mathrm{Buck}} = A\exp(-Br) - \frac{C_6}{r^6}, \tag{3.2}$$

where r is the distance between two particles and A, B, and C_6 are coefficients to describe the interactions. The TIP4P/2005 model uses the LJ potential

$$U_{\mathrm{LJ}} = \frac{C_{12}}{r^{12}} - \frac{C_6}{r^6}, \tag{3.3}$$

where the coefficient C_{12} describes the repulsive interactions.

In the developed model, the interactions between water and non-water molecules have been modeled using LJ potentials. To ensure optimal performance, the C_6 coefficients were obtained from the geometric mixing rule

$$C_{6,ij} = \sqrt{C_{6,ii}C_{6,jj}}, \tag{3.4}$$

where the indices i and j refer to particles of different types. As discussed below, there was a scenario in which we discarded the use of this mixing rule. However, the final optimized force field obeys geometric mixing for all dispersion coefficients. Aside from vdW interactions, the model uses fixed partial charges, bond, angle, and dihedral potentials. The complete model is given in Appendix B.

3.2.3 Fitting procedure

3.2.3.1 Nonbonded surfactant interactions

In this section we describe the modifications to the original polymer models resulting from enforcing geometric mixing rules on the C_6 dispersion coefficients for better compatibility with long-range dispersion solvers. We show that our modifications are minor and do not violate the methods applied in the original development. The reparameterization of these coefficients is justified for several reasons. First, the repulsive and attractive parts of the Buckingham potential have been determined independently for the polymer force fields that we are modifying [53, 54, 117, 118]. Second, the deviation from the geometric mixing rule is usually very small (except for C-F interactions). Moreover, determining the dispersion coefficients is the most challenging part in quantum chemistry-based force field development [117]; dispersion coefficients are thus subject to the largest uncertainties. In addition, geometric mixing for dispersion coefficients has also been enforced in parts of the original force fields [118].

For PDMS, no modifications were needed, as geometric mixing rules were already fulfilled. For PEO and PE, deviations from geometric mixing were very small, so geometric mixing was enforced for these compounds. For PTFE, the deviation from the mixing rule for interactions between C and F atoms was strong. To avoid inconsistencies between the C-C interactions in this and other models, the F-F and C-F dispersion coefficients were reparameterized as follows: two CF_4 molecules were repeatedly placed in random configurations. The F-F and C-F dispersion coefficients were reparameterized to ensure optimum agreement between the potential energy of the original and new models while obeying the geometric mixing rule. Afterwards, dispersion interactions were scaled to reproduce experimental quantities of surface tension, liquid density, and heat of vaporization of the model molecules hexane, dimethoxyethane (DME), and perfluorohexane. The scaling was required because all deviations from the geometric mixing rule in the original polymer force fields were such that the coefficients between unlike elements was larger than what would be obtained from the mixing rule. Thus, the enforced mixing reduced the net attraction between the molecules. This effect was counterbalanced by the applied scaling. Similar scaling has already been applied by the developers of the original polymer models in [117] to compensate for not using polarizability.

The repulsive part of the Buckingham potential was not parametrized for F-Si, F-O, and F-H interactions in the original force field. Here we use the Mason-Rice mixing rule [139]

$$A_{ij} = \sqrt{A_i A_j} \tag{3.5}$$

$$B_{ij} = \frac{B_i + B_j}{2} \tag{3.6}$$

to describe the missing parameters. This mixing rule has also been applied in parts of the original models [118].

3.2.3.2 Surfactant–water interactions

The usual approach in force field development is to start with the simplest type of interactions, constrain the optimized parameters, and then move to more complex systems. This would suggest to start the polymer-water parameterization with alkane–water mixtures. Here, however, we follow a different approach and fit the PEO–water interactions first. The other compounds considered here mix poorly with water, so there is little experimental data available. Moreover, the proper description of PEO–water interactions can be quite challenging [121, 140–143] and should therefore be addressed first.

Suitable parameters for the PEO–water interactions were determined from computations with DME and water. To fit the interactions, we follow three approaches. In the first approach, labeled as I, we fit the DME–TIP4P/2005 potential to a quantum chemistry-based potential for the DME–TIP4P model [140]. In that work, the DME–TIP4P interactions were fit to a series of different DME–water dimer configurations. Instead of redoing the quantum-chemistry computations, we fit the DME–TIP4P/2005 interactions to the original DME–TIP4P force field for a similar set of configurations of the DME–water dimer, so that the DME–water interactions are indirectly fit to quantum-chemistry computations. We also dropped the requirement of geometric mixing for the dispersion coefficients, because the quantum-chemistry data could not be reproduced otherwise.

In the second approach, labeled as II, we fitted LJ potentials to the Buckingham potential of the original force field, and defined the interaction parameters with water by applying standard Lorentz-Berthelot mixing rules to the fitted potential, as suggested by Ismail et al. [144]. In the third approach, labeled as III, we have simulated a series of DME–water mixtures with mole fraction $x_{\mathrm{DME}} \approx 0.18$. The parameters were fit to reproduce experimental values for the liquid density and viscosity. During the iterative fitting the radial distribution function was monitored to avoid demixing. In this approach, we applied geometric mixing for the dispersion coefficients. The LJ parameters for the interactions between the TIP4P/2005 hydrogens and the polymer compounds were set to zero in all three approaches.

In the next step, the LJ parameters describing the interaction between water and the polymeric compounds for the C, O, and H atoms were constrained. The remaining Si–water and F–water interactions were parametrized to match experimental values for the octamethyltrisiloxane (OMTS)–water and perfluorohexane–water interfacial tensions.

3.2.3.3 Partial charges and bonded surfactant interactions

The polyalkene, PDMS, PEO, and PTFE force fields were developed and published separately [53, 54, 117, 118]. Consequently, parameters describing links between the models are missing. Here we describe how these parameters were determined for the surfactants studied here.

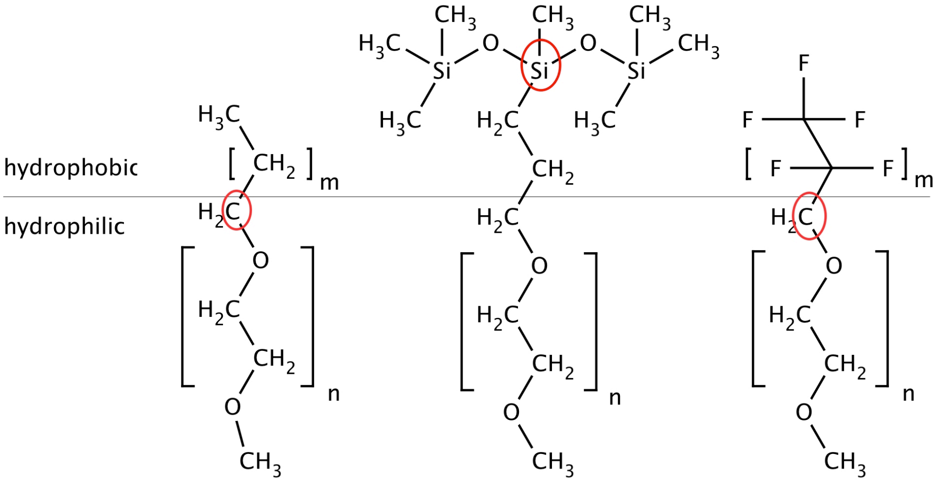


Figure 3.1: Alkyl ethoxylate (left), trisiloxane (center), and perfluoroalkane (right) surfactants. The hydrophilic part is the same for all surfactants. The encircled atoms were selected as “central” atoms of the surfactant for purposes of measuring the surface coverage and diffusion of the surfactants in Section 3.2.6. (Reprinted with permission from Ref. [145])

For the trisiloxane surfactant, depicted in Figure 3.1, several partial charges for the linker between the hydrophobic OMTS head group and the hydrophilic PEO head group were not available. The redefined parameters are the charges of the three CH_2 groups that link the central Si atom to the first O atom of the PEO chain. Moreover, parameters for the Si-C-C angle and the C-Si-C-C, O-Si-C-C, Si-C-C-C, Si-C-C-H, and O-C-C-C dihedral angles were not available in the literature. The missing parameters were determined from quantum-chemistry computations of a small version of the

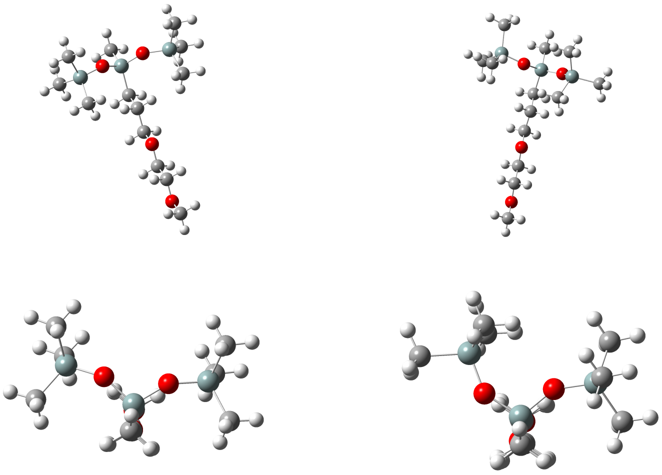


Figure 3.2: Symmetric (left) and unsymmetric (right) conformer of the trisiloxane surfactant. Bright gray: hydrogen atoms; dark grey: carbon atoms; red: oxygen atoms; dark teal: silicon atoms. In the lower images the surfactants have been rotated such that the differences in the head group are highlighted. (Reprinted with permission from Ref. [145])

trisiloxane surfactant with $n = 1$ (cf. Figure 3.1). All quantum-chemistry computations were performed with Gaussian 2009 [146].

We determined partial charges from a set of conformers of the trisiloxane surfactant. The starting point for the generation of conformers were two optimized structures of the trisiloxane surfactant. In both structures, all dihedral angles of the hydrophilic tail were in the trans configuration. For the first structure, both Si-O-Si-O dihedral angles are approximately 150°. For the second structure, one dihedral is approximately 160° whereas the other is 10°. The images of the conformers are depicted in Figure 3.2. New structures of these starting configurations were generated by performing 120° rotations around all dihedral angles in the hydrophilic tail. All possible combinations of trans and gauche and the two conformers for the hydrophobic head group were generated resulting in 256 structures.

These configurations were optimized at the HF/6-31g(2d) level. This is in agreement with the level of optimization used in the original force field [118]. Some of these optimizations failed because the generation of starting structures yielded configurations with strong overlaps. These configurations were excluded from further computations, leaving 237 configurations.

We applied the CHELPG algorithm [147] to the optimized structures to generate the missing partial charges. We used a cubic grid with spacing 0.3 . The excluded volume radii were $r_\mathrm{H} = 1.8\,\text{Å}$, $r_\mathrm{C} = 2.5\,\text{Å}$, $r_\mathrm{O} = 2.0\,\text{Å}$, and $r_\mathrm{Si} = 2.5\,\text{Å}$. In addition to overall charge neutrality, we also tried several other constraints: (1) setting the charges of all Si atoms, all Si and all O atoms, or all atoms for which the original model charges will be used in the final model to the values given in the literature; (2) enforcing equal charges on H atoms bonded to the same C atom; and (3) enforcing the sum of the newly defined charges to be equal to minus the sum of the charges of atoms that were not reparameterized from the original polymer models to ensure the final model is charge-neutral. All possible combinations of these constraints were applied. The charges for the final model were obtained from an unweighted average over the partial charges of all conformers. The charges used are those obtained by applying only constraint (3), because the charges obtained for the first and last CH_2 groups were in good agreement with literature values. Moreover, this constraint ensures charge neutrality in the final model and imposes, compared to (1), little bias. We also examined the effect of applying Boltzmann averages at 300 K instead of unweighted averages. The absolute difference in the resulting charges was less than 0.01 elementary charges for each atom. Using the weighted charges instead would have a negligible effect in the final model.

The potentials for angles and dihedrals not defined in the original polymer models were fit by bending and rotating around the unknown angle and dihedrals of an optimized trisiloxane structure. The energies for the perturbed structures were computed at the HF/6-31G(2d) level. Angle and dihedral potentials consistent with the functional form of the initial force field were fitted to the energies obtained from quantum-chemistry computations.

For the perfluoroalkane and alkyl ethoxylate surfactants, because of their weak polarity and also because, unlike the trisiloxane surfactant, the charges close to the linker are all near zero, the linking group charges were not redefined. For the alkyl ethoxylate surfactant, there are no missing bonded interaction parameters. For the perfluoroalkane surfactant, several dihedral parameters for the linker are not available. We fit those parameters as for the trisiloxane surfactant.

3.2.4 Force field validation with model molecules

Here we discuss some findings from our fitting procedure, as well as results obtained from the optimized force field. The complete set of parameters is given in Appendix B. For convenience, all parameters taken from the literature, including the bonded parameters, are also given there with references to the original sources.

3.2.4.1 Nonbonded surfactant interactions

Results for the modified nonbonded surfactant interactions are presented here. The model was altered only by enforcing geometric mixing in the dispersion coefficients and scaling them by two to three percent. Moreover, mixing rules were introduced for the repulsive parameters not explicitly defined. Because the modifications are rather small, these modifications are validated with a few tests only.

We ran simulations to obtain liquid densities ρ_{liq}, heats of vaporization $\Delta_{\text{vap}}H$, and surface tensions γ of the model molecules hexane, DME, perfluorohexane, and OMTS. Liquid densities and energies U_{liq} were determined from bulk simulations with $n_{\text{hexane}} = 574$, $n_{\text{DME}} = 719$, $n_{\text{perfluorohexane}} = 370$, and $n_{\text{OMTS}} = 261$ molecules. Simulations were executed in a cubic box with side length $L \approx 50$. To determine liquid densities, simulations were run in the NPT ensemble with a Nosé-Hoover barostat and thermostat [68,148,149] at 1 bar and 298 K. Simulations were equilibrated for 1 ns. Liquid densities were determined from subsequent 1 ns production runs.

To determine heats of vaporization, the protocol was similar to the density simulations, except for the temperatures, which depended on the experimental values used: $T_{\text{hexane}} = 342\,\text{K}$, $T_{\text{DME}} = 358\,\text{K}$, $T_{\text{perfluorohexane}} = 298\,\text{K}$, and $T_{\text{OMTS}} = 426\,\text{K}$. After equilibration, simulations were run for 3 ns in the NVT ensemble to measure the energy of the liquid U_{liq}. The energy of the gas phase U_{gas} was determined by simulating a single molecule with Brownian dynamics with equilibration for 50 ns and a 250 ns production phase. From these quantities, the heat of vaporization was computed from

$$\Delta_{\text{vap}}H = U_{\text{liq}} - U_{\text{gas}} + RT, \tag{3.7}$$

where R is the gas constant and T is the temperature.

Surface tensions were obtained from slab simulations in a box with dimensions $50 \times 50 \times 150$. The number of molecules in the simulations ($n_{\text{hexane}} = 689$, $n_{\text{DME}} = 863$, $n_{\text{perfluorohexane}} = 444$, and $n_{\text{OMTS}} = 313$) were chosen so that the thickness of the resulting slab is approximately 60 . The simulations were equilibrated in the NVT ensemble for 1 ns with a Nosé-Hoover thermostat at 298 K. The surface tension γ was determined as described in Appendix A from production runs over 5 ns.

For all simulations described above in this section, starting configurations were generated with PACKMOL [71]. Simulations were integrated with a Verlet integrator [83] and a 1 fs time step. The damping constants for the Nosé-Hoover barostats and thermostats were set to $\tau_p = 1000\,\text{fs}$ for the pressure and $\tau_T = 100\,\text{fs}$ for the temperature. PPPM with relative accuracy of 0.0001 has been used to compute long-range electrostatic interactions [34]. In bulk-phase simulations, dispersion interactions were truncated at 10 and long-range corrections applied. In surface simulations, PPPM for dispersion has been applied to accurately capture long-range dispersion effects. Uncertainties for the densities and heats of vaporization were determined from block averages. From the different charges for PDMS offered in [118], we used OMTS-specific charges.

Material	quantity	exptl[a]	original[b,e]	optimized[c,e]	Ref.[d]
hexane	ρ_{liq} [g/L]	654.96	644.43 (94)	650.56 (26)	[150]
	γ [mN/m]	17.91	17.89 (50)	18.72 (50)	[151]
	$\Delta_{\text{vap}}H$ [kcal/mol]	6.90	7.27 (6)	7.56 (10)	[152]
perfluorohexane	ρ_{liq} [g/L]	1679.8	1611.28 (384)	1648.61 (430)	[153]
	γ [mN/m]	12.23	12.91 (70)	13.57 (70)	[154]
	$\Delta_{\text{vap}}H$ [kcal/mol]	7.51	7.91 (3)	8.28 (8)	[155]
DME	ρ_{liq} [g/L]	861.3	855.47 (100)	859.59 (68)	[156]
	γ [mN/m]	23.93	27.43 (60)	27.52 (60)	[156]
	$\Delta_{\text{vap}}H$ [kcal/mol]	8.65	8.51 (3)	8.61 (2)	[157]
OMTS	ρ_{liq} [g/L]	815.5	815.33 (48)	-	[158]
	γ [mN/m]	16.6	19.28 (100)	-	[159]
	$\Delta_{\text{vap}}H$ [kcal/mol]	9.45	10.38 (2)	-	[160]

[a]Experimental values; [b]Simulated values using original force fields; [c]Simulated values using optimized force fields. (OMTS force field was not modified.) [d] Source of experimental data. [e]Values in brackets are twice the standard deviation of the mean obtained from the statistical uncertainty quantification

Table 3.1: Experimental and simulated quantities for model compounds. (Modified reprint with permission from Ref. [145])

Our simulation results are summarized in Table 3.1. Overall, the agreement between experimental and simulated values is good. For hexane, DME, and OMTS, the deviation in the liquid densities is less than 2% for the original and less than 1% for the modified model. For perfluorohexane the deviations are larger: 4% for the original model and 2% for the modified model. Deviations in the surface tension are better then 15% for all compounds for both models. For the heats of vaporization, the maximum deviation of the original model from experiment (10%) occurs for OMTS. For the modified model, deviations of similar size are also obtained for hexane and perfluorohexane. For these compounds, the deviation is only around 5% in the original model.

Compared to the original literature model, liquid densities are better reproduced for all compounds. The surface tensions and heats of vaporization are also in good agreement, although they are slightly worse than those predicted by the original force field. The performance of both models to describe the properties in Table 3.1 is comparable; neither model strongly outperforms the other for any of the observed quantities. An advantage of the modified version of the model, however, is that dispersion coefficients follow geometric mixing rules, and thus long-range dispersion solvers can be applied more efficiently. The computational performance of the modified model will thus be better in simulations in which the application of these solvers is desirable, such as simulations of interfacial or strongly inhomogeneous systems. Because further increasing the dispersion coefficients would increase all of the quantities reported here, the resulting parameters are a trade-off between accuracy in the structural and energetic properties. Further optimization would have been possible either by modifying the repulsive parameters or by individually adjusting the dispersion coefficients. However, this would complicate extending the model, because the deviation from the original development would be greater. Moreover, as the agreement between experiment and simulation is sufficient for our purpose, we have refrained from further optimization.

Simulations of mixtures of hexane and perfluorohexane like those described in [161] were used to validate the mixing rules introduced for the missing repulsive interaction coefficients in Equations (3.5) and (3.6). These mixtures were chosen because properly describing them is challenging and typically requires selection of special mixing rules or modifications of the potentials [55, 161]. Simulations have been performed at 1 bar and 298 K. Simulations were run in the *NPT* ensemble with a Nosé-Hoover thermostat and barostat, with 1 ns equilibration and 2 ns sampling, during which we measured the density and the enthalpy. The number of molecules was chosen such that the box length of the cubic boxes is approximately 50 in the equilibrated state. Simulations were performed at molar fractions $x_{\mathrm{C_6F_{14}}} \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$ with $n_{\mathrm{C_6F_{14}}} \in \{0, 125, 222, 300, 370\}$ $\mathrm{C_6F_{14}}$ molecules and $n_{\mathrm{C_6H_{14}}} \in \{575, 375, 222, 100, 0\}$ $\mathrm{C_6H_{14}}$ molecules.

The excess volumes V_{e} and enthalpies of mixing $\Delta_{\mathrm{mix}}H$ were determined from

$$V_{\mathrm{e}}(x_{\mathrm{C_6F_{14}}}) = V(x_{\mathrm{C_6F_{14}}}) - (x_{\mathrm{C_6F_{14}}} * V_{\mathrm{C_6F_{14}}} + (1 - x_{\mathrm{C_6F_{14}}})V_{\mathrm{C_6H_{14}}}) \tag{3.8}$$

and

$$\Delta_{\mathrm{mix}}H(x_{\mathrm{C_6F_{14}}}) = H(x_{\mathrm{C_6F_{14}}}) - (x_{\mathrm{C_6F_{14}}} * H_{\mathrm{C_6F_{14}}} + (1 - x_{\mathrm{C_6F_{14}}})H_{\mathrm{C_6H_{14}}}), \tag{3.9}$$

where $x_{\mathrm{C_6F_{14}}}$ is the molar fraction of perfluorohexane, $V_{\mathrm{C_6F_{14}}}$, $V_{\mathrm{C_6H_{14}}}$, $H_{\mathrm{C_6F_{14}}}$, and $H_{\mathrm{C_6H_{14}}}$ are the molar volumes and enthalpies of pure perfluorohexane and hexane, and $V(x_{\mathrm{C_6F_{14}}})$ and $H(x_{\mathrm{C_6F_{14}}})$ are the molar volumes and enthalpies at $x_{\mathrm{C_6F_{14}}}$.

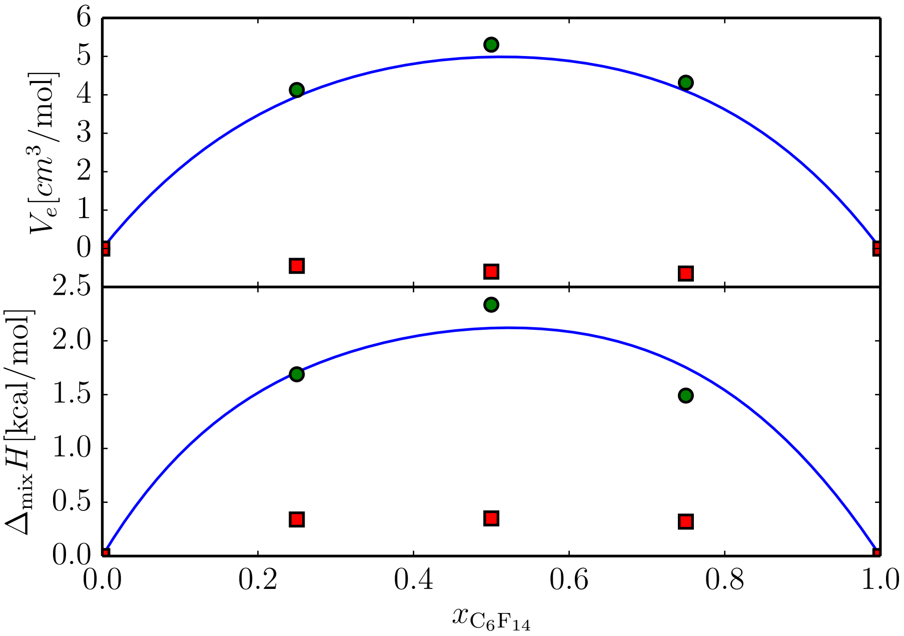


Figure 3.3: Excess quantities for mixtures of perfluorohexane and hexane. Top: Excess molar volume, bottom: enthalpy of mixing. Blue lines: Experimental fits taken from [162] for excess volume and from [163] for heat of mixing. Red squares: simulated values for OPLS from [161]. Green circles: simulated values using model presented here. (Modified reprint with permission from Ref. [145])

The results of the simulations are shown in Figure 3.3. The agreement between experiment and simulation is not perfect, but still significantly better than that obtained with the OPLS force field in [161]. We note that in [161] the deviations were not attributed to the OPLS model, but to the challenge of modeling alkane–perfluoroalkane mixtures. The good agreement between experiment and simulation with the quantum chemistry-based force field is mainly a result of the methodology used for the original force field, and only to a small extent a result of our modifications.

3.2.4.2 Surfactant–water interactions

In this section we briefly present results from the determination of interactions between the TIP4P/2005 model and our surfactant models. We start by giving results for the DME/water interactions before presenting results obtained for the other systems.

We ran simulations of DME–water mixtures at different molar fractions $x_{\text{DME}} \in \{0, 0.04, 0.1, 0.18, 0.27, 0.41, 0.64, 1.0\}$ with $n_{\text{DME}} \in \{0, 125, 200, 350, 475, 550, 650, 719\}$ DME molecules and $n_{\text{H}_2\text{O}} \in \{4123, 3053, 1858, 1616, 1279, 776, 361, 0\}$ water molecules. The range of concentrations was motivated by the simulations reported in [140]. We determined the liquid densities ρ_{liq}, viscosities η, diffusivities of water D_{w}, and radial distribution functions (RDF) $g(r)$ between the water O and DME O atoms. The number of molecules was chosen such that the cubic box length was approximately 50 Å after equilibration in the NPT ensemble at 1 bar and 318 K for 1 ns. Afterwards, densities were determined from simulations of at least 2 ns. Viscosities and diffusion coefficients were obtained from NVT simulations only for potential III (cf. Section 3.2.3.2). Viscosities and diffusion coefficients were determined using 4 ns simulations; RDFs were determined from 2 ns simulations for potential III. Because of the inaccurate densities obtained with potentials I and II, viscosities and diffusion coefficients were not computed based on these potentials. RDFs for these potentials were determined from trajectories of only 0.5 ns, which is sufficient to obtain qualitatively correct results. The water geometry was kept rigid using the SHAKE algorithm [77]. All other settings were the same as described above.

To determine the diffusion coefficients, we tracked in each simulation the movement of 10 randomly selected water molecules. Ten molecules were selected because this is a large enough sample to obtain meaningful averages while still keeping the amount of data written to disk during the simulation reasonable. The diffusion coefficients were computed from the mean square displacements (MSDs) of the water molecules using the Einstein relation

$$D = \frac{1}{2d} \lim_{t \to \infty} \frac{\langle \mathbf{r}(t) - \mathbf{r}(0) \rangle}{t}, \tag{3.10}$$

where d is the dimensionality of the system, $\mathbf{r}$ is the center of mass of a water molecule, and t is the time. MSDs with reduced noise were computed by selecting each time step of the stored trajectories as the starting point to compute a different MSD. The MSD for each molecule was obtained from averaging over the MSDs obtained with different starting points. Because the long-time limit $t \to \infty$ cannot be reached in finite-time MD simulations, the diffusion coefficients are determined from linear fits to the diffusive regime of the MSDs. We determined this regime separately for each tracked water molecule by identifying the regime with a slope of approximately unity in log–log representation of the MSD over time.

The viscosities η were determined from the Green-Kubo relations, which for isotropic systems are written as [164]

$$\eta = \frac{V}{10kT} \int_0^\infty \sum_{\alpha\beta} \langle P_{\alpha\alpha}(0) P_{\alpha\beta}(t) \rangle dt, \tag{3.11}$$

where V is the volume, k is the Boltzmann constant, α and β are Cartesian indices, and

$$P_{\alpha\beta} = \frac{1}{2}(p_{\alpha\beta} + p_{\beta\alpha}) - \frac{\delta_{\alpha\beta}}{3}\sum_{\gamma} p_{\gamma\gamma}, \tag{3.12}$$

where $p_{\alpha\beta}$ is the $\alpha\beta$-component of the pressure tensor, δ is the Kronecker delta, and γ is a Cartesian index. Because noise in the data can influence the long-time limit $t \to \infty$ in Equation (3.11), we evaluated the integral for different values of t. Before the data are influenced by noise simulation uncertainties for large values of t, the integral reaches a plateau. The viscosities reported below correspond to the height of this plateau.

Experimental liquid densities and values obtained from simulations with potentials I, II, and III (cf. Section 3.2.3.2) are given in Figure 3.4. For the potential from approaches I and II, the agreement between experiment and simulation is insufficient. For intermediate concentrations, the deviation between I and experiment is more than 2.5%, and for II the deviation is almost 6%. In approach III, where we fit the DME–water potential to experimental data, the liquid density is reproduced well for all concentrations, with deviations between simulation and experiment less than 0.5%.

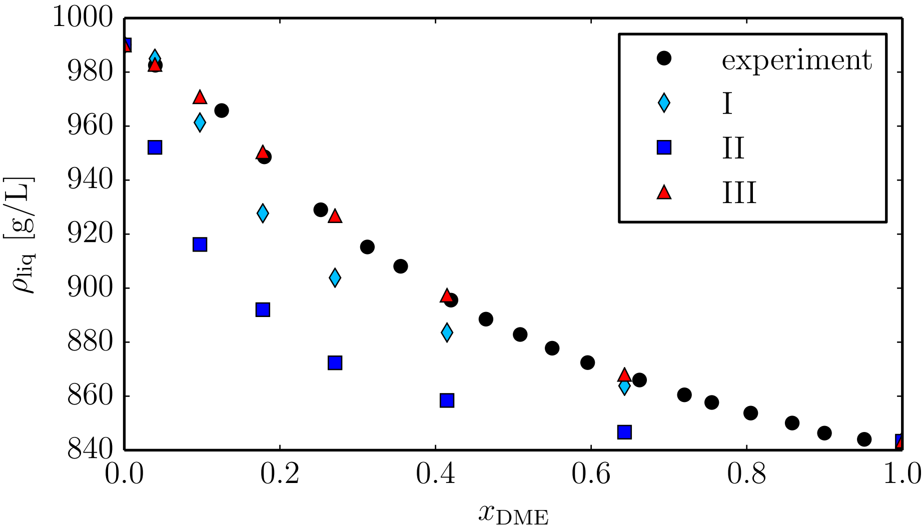


Figure 3.4: Experimental and simulated densities of DME–water mixtures at 318 K with model approaches I, II, and III. Experimental values were taken from [165]. (Modified reprint with permission from Ref. [145])

The underprediction of the densities by potentials I and II is related to partial or full demixing. As can be seen from the RDFs depicted in Figure 3.5, the first peak in the RDFs is distinctly below 1.0 for potential I and less than 0.25 for potential II, indicating demixing. Similar demixing in DME/water mixtures has already been detected by Fischer et al. [121] with other force fields for water and DME. For potential

III, however, the compounds mix properly. The correct phase behavior is also confirmed by the free energy of solvating a DME molecule in water, which we have computed using thermodynamic integration [166]. The integration was performed along three paths. In the first and second paths, a single DME molecule was dissolved in water. The DME–water interactions were turned off by gradually switching off the charges of the DME molecule in the first path, and switching off the vdW interactions between DME and water in the second path. In the third path, the charges of a single DME molecule in vacuum were switched on again. Simulations of the first and second path were equilibrated in the *NPT* ensemble for 0.5 ns with subsequent production runs for 1.5 ns in the *NVT* ensemble. The third path was equilibrated for 50 ns with subsequent production for 200 ns using Langevin dynamics [167]. The integration variable λ was adjusted from 0 to 1 in increments of 0.1 along each path. The given uncertainty is twice the root of the sum of the squares of the standard deviations of the mean obtained at each state point. The simulated value is $\Delta_{\text{sol}}G_{\text{sim}} = -5.4 \pm 0.3\,\text{kcal/mol}$, the experimental value is $\Delta_{\text{sol}}G_{\text{exp}} = -4.8\,\text{kcal/mol}$ [168].

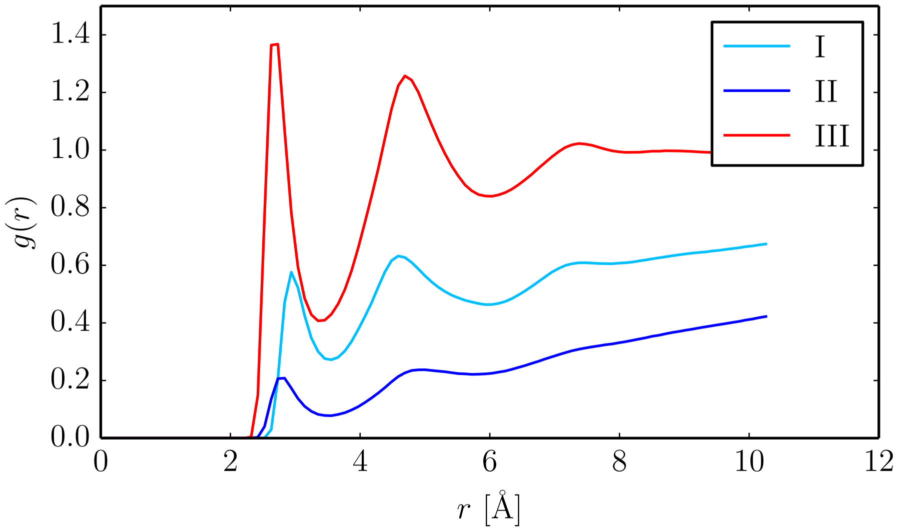


Figure 3.5: Radial distribution functions between the DME O and water O atoms for $x_{\text{DME}} \approx 0.18$. The low value of the first peak for potentials I and II indicates demixing. (Modified reprint with permission from Ref. [145])

Because of demixing and inaccurate reproduction of the density for potentials I and II, dynamic quantities were only determined for potential III and are depicted in Figure 3.6. For viscosities of the pure compounds, the deviation between simulation and experiment is around 7% for pure water and around 5% for pure DME. In the region of maximum deviation at $x_{\text{DME}} \approx 0.2$, the simulated value is around 14% higher than the experimental value, indicating that the dynamics are somewhat too slow in this region. For the diffusion coefficients, aside from the original experimental data from [142], this figure includes scaled values of the experimental data, as suggested in [121]. For the experimental data, it is unclear whether the original or scaled data better represent the true value. The simulated diffusion coefficient is often below the experimental value. Yet, the general trend that the diffusion coefficient passes through a minimum is reflected in the simulation results. Combining the results for the viscosities and the diffusion coefficients, it seems that the dynamics of the developed model is somewhat too "slow" compared to reality. However, the agreement is sufficient for our purposes.

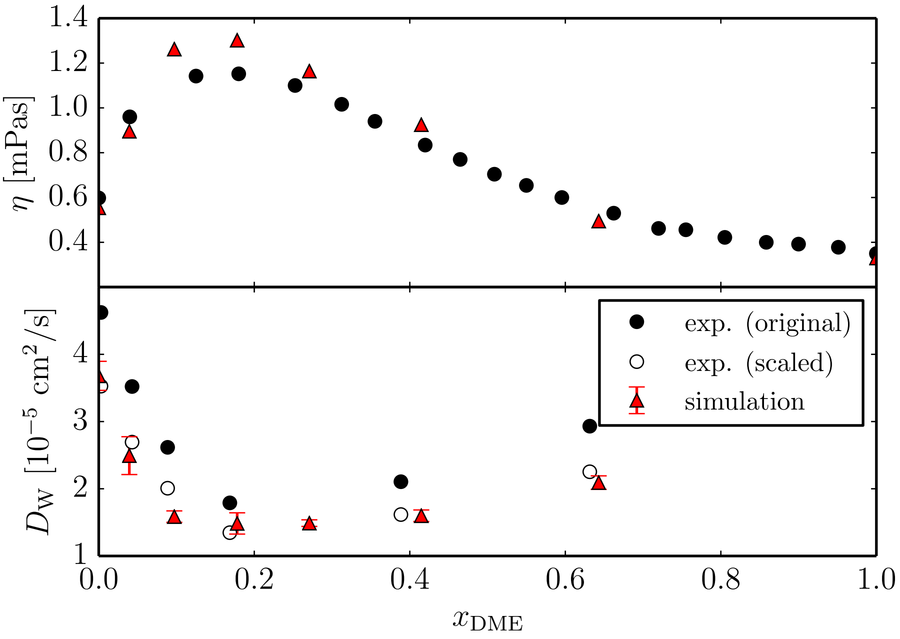


Figure 3.6: Experimental and simulated values for the viscosity (top) and water diffusivities (bottom) of DME–water mixtures. Experimental viscosities were taken from [165]. Original diffusivities were taken from [142], scaled diffusivities from [121]. (Modified reprint with permission from Ref. [145])

Simulations of the surface tension between water and the other model compounds were performed in elongated boxes. The number of water molecules in the simulation cell was $n_{\mathrm{water}} = 5000$. The number of model molecules was equal to the number of molecules in the surface simulations described in Section 3.2.4.1. The simulations were equilibrated for 1 ns, during which time, the box dimensions parallel to the interface was held fixed, while the other dimension was controlled with a Nosé-Hoover barostat to maintain a pressure of 1 bar. The temperature was fixed at 298 K with a Nosé-Hoover thermostat. Afterwards, NVT simulations were run for 5 ns. During this period, data was collected to compute the surface tensions. The results of the simulations are given in Table 3.2. The agreement between simulated and experimental data is good for all three compounds. We note that no additional fit was required for the hexane–water interactions.

	γ [mN/m]		
Material	experimental	simulation	Ref.
hexane	50.38 (4)	52.03 (120)	[169]
perfluorohexane	57.20(13)	57.94 (140)	[170]
OMTS	40.5 (5)	40.3 (18)	[171]

Table 3.2: Experimental and simulated interfacial tensions of model compounds and water. The last column contains references to the experimental data. (Modified reprint with permission from Ref. [145])

3.2.5 Discussion of the fitting strategies

In this section we discuss the three fitting strategies I, II, and III (cf. Section 3.2.3.2) and provide explanations for the performance of the models obtained with the different approaches.

In approach I we fit the DME–TIP4P/2005 potential to a previously developed quantum chemistry-based DME–TIP4P model [140]. Since the model obtained with this strategy is an indirect fit to quantum chemistry data, one might expect it to provide best results. While the original potential was successful in describing the mixture [140], the resulting model in this work provided a poor description of the system and resulted in demixing. Since we did not fit directly to quantum chemistry data, and since the potential upon which it was built was created with methods that were state of the art in 2002, one might argue that directly fitting to quantum chemistry data with more elaborate methods will improve the results. While we cannot guarantee that such an approach will not provide any benefits, we doubt that this approach will result in a model that will perform better than the model that was obtained with approach III in this work.

Aside from the potentially insufficient quantum chemistry data approach I was built upon, the reason for wrong mixing behavior lies within the TIP4P/2005 model itself. While the TIP4P/2005 model is known to provide good results for many observables, such as liquid densities, surface tensions, and diffusivities [120], its ability to reproduce microscopic phenomena is limited. Kiss and Baranyai showed that the TIP4P/2005 systematically underestimates the energy of formation for small water clusters [172]. Thus, the attraction between the TIP4P/2005 water molecules is too strong compared to real water or accurate quantum chemistry computations. To ensure mixing of DME with water, we must ensure that the DME–water interactions are sufficiently favorable to separate water molecules. Thus, these interactions will also have to be stronger than DME–water interactions in reality to compensate the strong interactions of the TIP4P/2005 model. This cannot be achieved with a fit to quantum-chemistry data.

Approach II, in which the potential of the mixture was determined from fitting LJ potentials to the Buckingham potentials used for the polymer models, yielded phase behavior even worse than that found using Approach I. Aside from too strong interactions between TIP4P/2005 molecules, there was was an additional problem related to the functional forms of the Buckingham and LJ potentials. We used a simple least-squares fit with the additional constraint that the C_6 coefficients in the final model obey geometric mixing rules. As a result, the minima in the LJ potentials were shallower than the minima in the Buckingham potentials. Specifically, the minima in the LJ potentials were $\epsilon_{\mathrm{O,LJ}} = 0.176\,\mathrm{kcal/mol}$, $\epsilon_{\mathrm{C,LJ}} = 0.041\,\mathrm{kcal/mol}$, and $\epsilon_{\mathrm{H,LJ}} = 0.003\,\mathrm{kcal/mol}$, whereas the minima in the Buckingham potentials were significantly larger: $\epsilon_{\mathrm{O,LJ}} = 0.206\,\mathrm{kcal/mol}$, $\epsilon_{\mathrm{C,LJ}} = 0.098\,\mathrm{kcal/mol}$, and $\epsilon_{\mathrm{H,LJ}} = 0.010\,\mathrm{kcal/mol}$. When applying mixing rules to these reduced potential depths, the resulting ϵ's describing the DME–water interactions were too small and caused the strong demixing. An alternative strategy to match the Buckingham and LJ potentials, for example, by dropping the constraint of geometric mixing or constraining the depth of the potentials, might have improved the results, but the problem of the strong interactions among TIP4P/2005 water molecules remains.

In approach III we fit the DME–TIP4P/2005 interactions such that macroscopic observables are reproduced well. This strategy automatically compensates for the strong interactions of the TIP4P/2005 model and in this way provides correct mixture densities and a good estimate for the free energies of solvation $\Delta_{\mathrm{sol}}G$. The dynamics obtained with this approach, however, were too slow compared to experiment. For an effective model such as the one generated with approach III it is difficult to make precise statements why it has certain features—for instance, why the dynamics are not reproduced better. An explanation could be that the resulting DME–water interactions of this fit are too strong compared to the DME–DME interactions and that the connections between DME and water molecules are therefore too stable.

3.2.6 Model validation with surfactant simulations

In previous sections, the developed force field was only applied to small model compounds. It is thus not clear whether the force field will be applicable to simulations with surfactants and characterize them sufficiently well. The validation of the developed model for the description of surfactants is presented here with simulations of surfactants at the water interface. A comparison to experimental data requires two features. First that reliable data is available, and second that the specific quantity is accessible from MD simulations. Comparison to experiment is therefore restricted to maximum surfactant loads and surface tensions at fully covered surfaces. The remaining data presented in this section only provide information that the model provides the correct qualitative behavior, such as that surfactants reside at the interface.

3.2.6.1 Simulations setup

We have performed a series of surfactant simulations with the alkyl ethoxylate, trisiloxane, and perfluoroalkane surfactants depicted in Figure 3.1. The chain length of the hydrophilic part was $n = 6$ for all surfactants, which corresponds to experimental observations of superspreading with the trisiloxane surfactant. Simulations with the alkyl ethoxylate surfactant were performed with $m = 10$ for the hydrophobic part. Simulations with the perfluoroalkane based surfactant were performed with $m = 8$.

The box dimensions were $60\,\text{Å} \times 60\,\text{Å} \times 200\,\text{Å}$. The number of water molecules was 10 000 in all simulations. The number of surfactants at each interface were $n_\text{S} \in \{0, 18, 36, 54, 60, 72, 78, 84\}$ corresponding to interfacial areas of $A_\text{mol} \in \{\infty, 200, 100, 66.67, 60, 50, 46.15, 42.85\}^2$/molecule and concentrations $\Gamma \in \{0, 0.5, , 1.0, 1.5, 1.67, 2.0, 2.17, 2.33\}\,\text{nm}^{-2}$. For the trisiloxane surfactant, simulations were not performed with $n_\text{S} = 78$ or $n_\text{S} = 84$ because the surface was already overcrowded for $n_\text{S} = 72$. Starting configurations were generated with PACKMOL [71]. A slab of water molecules was generated in the center of the simulation box with interface parallel to the xy plane. The surfactant molecules were placed so that their hydrophobic parts were at the water surface when the simulations began. A Nosé-Hoover thermostat [68] was used to keep the temperature at 298 K. The simulations were equilibrated for 1 ns. Afterwards, data was taken from production runs over 7 ns. The SHAKE algorithm [77] was used to constrain the shape of the water molecules. Long-range electrostatic and dispersion interactions were computed with the PPPM method with analytic differentiation [34], with parameters set according to Section 2.8. The equation of motion was integrated with the rRESPA algorithm [83, 84], with settings chosen as described in Section 2.8.

3.2.6.2 Surface crowd

We first examine the surfactant crowd at the interface, as it is possible to start simulations in an unphysical state in which the surface is overloaded with surfactants. Because of the finite length of the simulations, there is the danger that the system cannot escape from this starting configuration.

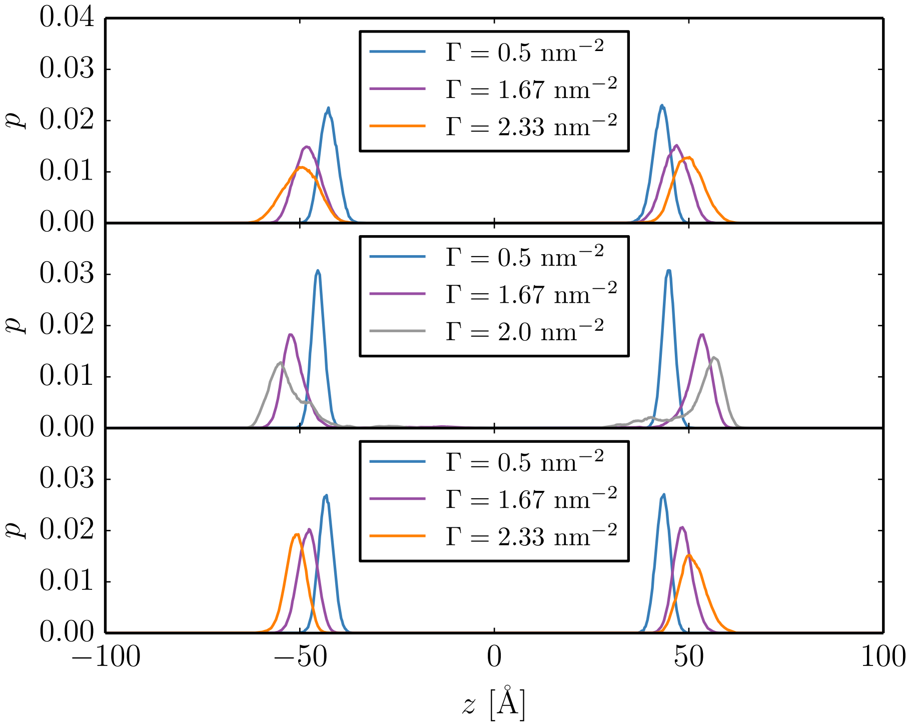


Figure 3.7: Histograms of the z position of the central atom (cf. Figure 3.1). For the trisiloxane surfactant with $\Gamma = 2.0\,\mathrm{nm}^{-2}$, the curve shows secondary peaks, indicating overcrowding. (Modified reprint with permission from Ref. [145])

As an indirect measure for whether the surface was overcrowded we have examined the positions of the central surfactant atoms in our simulations. Histograms of their positions are shown in Figure 3.7 for selected values of Γ from which the differences between the simulations become apparent. For the alkyl ethoxylate and the perfluoroalkane surfactant the only visible effect in the range of examined surfactant loads is that the distribution broadens and the height of the peak is reduced. Moreover, the peaks at the left and right interfaces are not identical. The peaks have slightly different heights at high surfactant concentration, indicating that further sampling may be required to obtain more accurate values. For the trisiloxane surfactant, however, the situation is different. For low and intermediate concentrations, the behavior is the same as for the alkyl ethoxylates and perfluoroalkanes. At the highest concentration, however, the profile is much noisier, and the histogram shows secondary peaks, indicating that the system was in an unphysical, overcrowded state. Moreover, when visualizing the trajectories, bending of the interface could be observed at the highest surfactant load, indicating that additional interfacial area is required to hold the surfactants. This bending was not observed in any other simulation. Although this simulation was in an unphysical state, simulation results are reported for completeness.

3.2.6.3 Density profiles

Experimental density profiles to which simulated data could be compared to are not available. Yet, simulated density profiles can at least provide information whether the expected qualitative behavior is captured successfully in the simulations.

Number density profiles ρ_{num} of water and the different surfactant parts are depicted in Figure 3.8. The absolute value of the z dimensions between simulations with a different number of surfactants contains very limited information. The number of water molecules is identical in all simulations; thus, with increasing number of surfactants, the system grows and the entire interfacial region moves. The z positions of the density peaks move not only because the interface broadens, but also because the interface itself moves. For fixed surfactant load, the absolute position of the z value is of course reasonable quantity for comparisons.

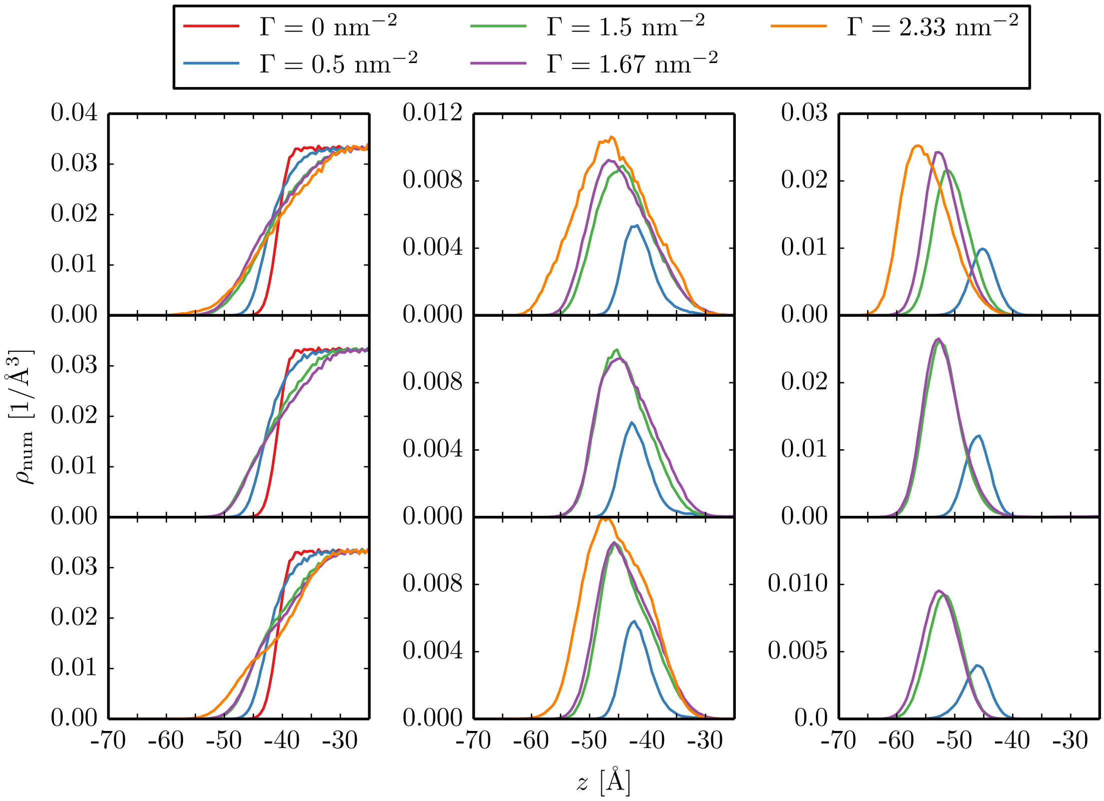


Figure 3.8: Number densities as a function of z. Rows from top to bottom: Alkyl ethoxylate, trisiloxane, and perfluoroalkane surfactant. Columns from left to right: Water O, hydrophilic O, and hydrophobic atoms. (Modified reprint with permission from Ref. [145])

The interface widens with increasing surfactant concentration for all three examined surfactants. While the thickness of the interface is only approximately 5 Å for pure water, the total width of the interface including the hydrophobic part of the surfactant becomes as large as 35 Å for the surfaces covered with alkyl ethoxylate and perfluoroalkane surfactants and 30 Å for trisiloxane surfactants at maximum coverage. Even for the smallest examined surfactant concentration $\Gamma = 18\,\mathrm{nm}^{-2}$, the width of the interface increases to between 15 Å and 20 Å.

The number densities of the water O atoms are shown on the left of Figure 3.8 for the different surfactants. In each case, the distribution broadens with increasing surfactant concentration. It is noteworthy, though, that the change of the curves is especially strong for low surfactant concentrations. For higher concentrations ($\Gamma \geq 1.5\,\mathrm{nm}^{-2}$) the profile does not broaden significantly.

The distribution of the hydrophilic O atoms depicted in the middle row of Figure 3.8 becomes broader and higher with increasing surfactant concentration. For low concentrations, the distribution is rather narrow compared to higher concentrations for all three surfactants. This indicates that at low concentrations, the hydrophilic part of the surfactant resides close to the interface. With increasing surfactant concentration, the interface narrows and the hydrophilic tail stretches out into the water phase.

The extent of surfactant hydration is apparent from a comparison of the distribution of the water molecules and the hydrophilic O atoms. For low surfactant concentrations ($\Gamma \leq 1.5\,\mathrm{nm}^{-2}$), the distributions decay to zero at approximately the same value of z, which means that the hydrophilic parts are fully hydrated. For higher surfactant concentrations ($\Gamma = 2.33\,\mathrm{nm}^{-2}$), however, the distribution of water molecules decays to zero more quickly, meaning that some of the hydrophilic O atoms stick out of the water surface. At high concentrations, the hydrophilic parts of the surfactants are only partly hydrated.

The distribution of the hydrophobic atoms are depicted in the right column of Figure 3.8. Like the hydrophilic part, the distribution is narrow for low surfactant loads, but broadens with increasing concentration. For the alkyl ethoxylate and perfluoroalkane surfactants, this is the result of the hydrophilic chains starting to lift away from the interface and stretching into the vapor phase.

All surfactants resided at the interface with the hydrophilic parts of the surfactants hydrated and mixing with water atoms, while the hydrophobic parts were not. This matches the expected behavior from experiment. Of course this finding does not necessarily imply that the developed model accurately represents the different surfactants, but it does show that the most basic features of surfactants are reproduced.

3.2.6.4 Surface tensions and maximum coverage

Simulated and experimental values for the surface tension at different surfactant concentrations for the free and fully covered surfaces are depicted in Figure 3.9. For the highest surfactant concentrations in simulations that are not overcrowded ($\Gamma = 2.33\,\mathrm{nm}^{-2}$ for the alkyl ethoxylate and perfluoroalkane surfactants, and $\Gamma = 1.67\,\mathrm{nm}^{-2}$ for the trisiloxane surfactant), the estimated surface tensions agree with experimental values at fully crowded surfaces. For the area per molecule at maximum packing $A^{\infty}_{\mathrm{mol}}$, however, there are deviations between modeling and experiment, as shown in Table 3.3. While for the trisiloxane surfactant the result from the simulation is in between the values obtained from macroscopic models, $A^{\infty}_{\mathrm{mol}}$ is much smaller for the alkyl ethoxylate surfactant compared to macroscopic models and experiment. The deviations might result either from the experiments and modeling not being sufficiently accurate, or that the state of maximum packing should be attributed to lower values of the number of surfactants at the interfaces in our simulations. In the latter case our model would not reproduce well the reduction of the surface tension for the alkyl ethoxylate surfactant. There is no indication from our simulations, however, that sug-

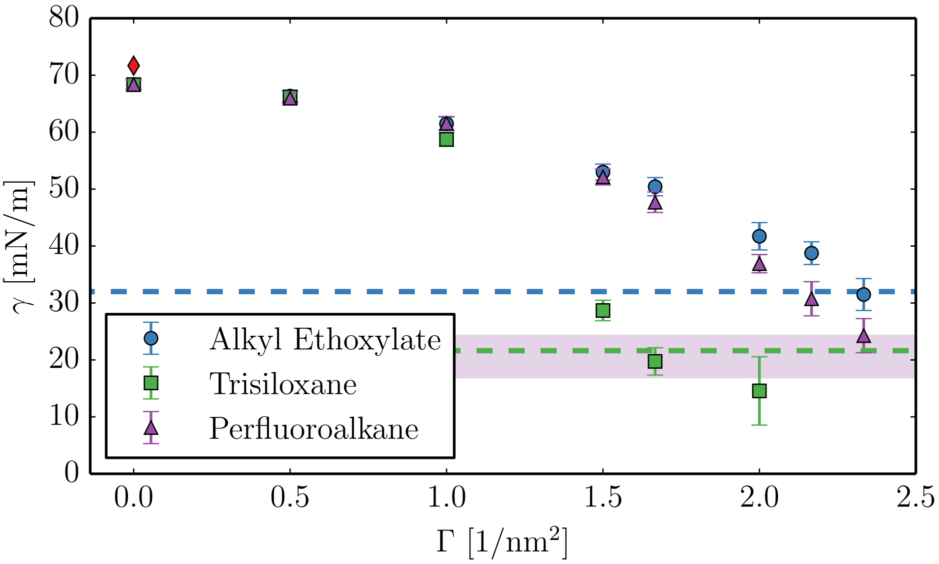


Figure 3.9: Simulated and experimental surface tensions at 298 K. Red diamond: experimental value for pure water [173]. Other scatter points: simulated values. Dashed lines: experimental values for fully covered surfaces [174, 175]. Experimental data for the alkyl ethoxylate surfactant were measured for a slightly different surfactant with $m = 11$ (cf. Figure 3.1) and hydroxyl end cap. Shaded area: range of experimental values at fully covered surfaces for slightly different perfluoroalkane surfactants with $m = 3$–5 and $n = 3$–5 (cf. Figure 3.1) [176–178]. (Modified reprint with permission from Ref. [145])

gests that simulations with either the alkyl ethoxylate or perfluoroalkane surfactant with $\Gamma = 2.33\,\mathrm{nm}^{-2}$ were overcrowded. Also, there is no obvious explanation why the area per molecule of a chain molecule like the alkyl ethoxylate should occupy the same area per molecule as the bulky trisiloxane surfactant, as suggested by the models and measurements. Given that small differences in the end caps of large molecules can have strong effects [176, 179–183], the deviations in Table 3.3 might also result from the maximum packing for the reference data being determined from surfactants with a hydroxyl end group, whereas the surfactants in our simulations have a methyl end group.

surfactant	$A^{\infty}_{\mathrm{mol,Neut}}$ [Å^2]^a	$A^{\infty}_{\mathrm{mol,Lang}}$ [Å^2]^b	$A^{\infty}_{\mathrm{mol,Frum}}$ [Å^2]^c	$A^{\infty}_{\mathrm{mol,Sim}}$ [Å^2]^d
trisiloxane^e	-	70.6^i	54.3^i	60
alkyl ethoxylate^f	55 ± 3^h	68^j	56^j	43
perfluoroalkane^g	-	-	-	43

^{a}obtained from neutron reflectivity measurement; ^{b}obtained from a Langmuir model; ^{c}obtained from a Frumkin model; ^{d}estimated from our simulations; ^{e}trisiloxane surfactant in experiment had a hydroxyl end group; ^{f}alkyl ethoxylate surfactant in experiment had a hydroxyl end group and the hydrophobic chain contained one additional CH_2 group; ^{g}no reference data available; ^{h}taken from [184]; ^{i}taken from [185]; ^{j}taken from [175];

Table 3.3: Surface area per surfactant molecule at the interfaces at maximum packing (Reprinted with permission from Ref. [145])

3.2.7 Assessment of the developed force field

The developed force field reproduces quantities important for interfacial simulations, such as surface tensions, free energies, and structural properties [127] of simple model molecules with reasonable accuracy. Dynamics are also reproduced reasonably well, but not as accurately as the properties mentioned before. Moreover, a strong feature is that the parameters for the non-water molecules are based on quantum-chemistry computations. It was shown that this ensures transferability even for the difficult problem of mixtures of alkanes and perfluoroalkanes. The force field can therefore be extended to further compounds using the methods employed in the original development. Moreover, the detailed quantum chemistry-based force field has the advantage of capturing features specific to the studied molecules, such as the flexible Si-O-Si angle for OMTS [118], which might be relevant to the superspreading problem.

A disadvantage of the model is that reasonable agreement between simulated and experimental quantities for mixtures with water could only be achieved by fitting parameters to reproduce experimental quantities, which limits transferability compared to the non-water interactions. The difficulty of combining models developed with different strategies—the quantum chemistry-based non-water models and the highly empirical TIP4P/2005 water model—is a general problem in force field development. It should be noted though, that this problem is more or less pronounced for different models. For example, a quantum chemistry-based force field to describe the interactions of DME with the original TIP4P model was successful in describing properties of the mixture [140]. Apparently the capacity of TIP4P/2005 to reproduce a variety of quantities of pure water [120] is achieved at the expense of difficulties when combined with other models. Yet, because of its accurate representation of surface properties and computational simplicity, it is still our model of choice.

The validation of the model in simulations of surfactants at water interfaces provided satisfying results. Aside from the surface free energies, which are important for wetting phenomena, being reproduced well by the developed model, the model qualitatively provides the correct hydration behavior. These results, in combination with the validation of the force field for simple molecules, suggest that the developed model accurately represents relevant properties of superspreading surfactants and will realistically model the spreading phenomena described in Section 3.3. The force field could also be used as a starting point for the development of coarse-grained potentials for trisiloxane surfactants.

3.3 Simulations of surfactant enhanced spreading

3.3.1 Simulation setup and analysis methods

3.3.1.1 Simulation setup

We performed MD simulations with cylindrical droplets of aqueous solutions of different surfactants on various polymer substrates. The substrate was parallel to the xy plane and the axis of the cylinder was in the y direction in our setup. Simulations of cylindrical droplets are computationally less intensive than those of spherical droplets and avoid effects caused by line tension. However, some properties, such as the spreading exponent, depend on the droplet geometry and have to be translated between simulation and experiments that are usually performed with spherical droplets [186].

Simulations were performed with three different polymer substrates: polypropylene (PP), PEO, and PTFE. These different substrates include a substrate that is too hydrophobic (PTFE), one that is just right (PP) [187], and another that is too hydrophilic (PEO) for superspreading. Spreading experiments with trisiloxane-laden water droplets on PEO are not available, yet, the small contact angle of around 20° indicates that the material is too hydrophilic for superspreading [3, 188]. To prepare the substrates, monodisperse polymer chains with $n_{\mathrm{mon,PP}} = 101$, $n_{\mathrm{mon,PEO}} = 50$, or $n_{\mathrm{mon,PTFE}} = 101$ monomers were used to create amorphous bulk configurations with the Polymer Builder of the MAPS platform [189]. The size of the box was $50\text{Å} \times 80\text{Å} \times 30\text{Å}$. The number of chains were $n_{\mathrm{PP}} = 30$, $n_{\mathrm{PEO}} = 34$, and $n_{\mathrm{PTFE}} = 31$. The bulk systems were equilibrated in the NPT ensemble at a pressure of 1 atm with variable box dimensions only in the z-direction. The simulation was run for 1 ns at 500 K, then cooled down to 300 K within 1 ns and run for another nanosecond at 300 K. Afterwards, periodic boundary conditions were removed in the z-direction, the box length in z-direction was extended and the simulation was run for another 1.5 ns. The resulting slabs had a thickness of approximately 30 Å and were used to generate substrate materials for the spreading simulations. Simulations setups with different substrates are referred according to the abbreviated polymer name below.

We ran simulations with trisiloxane surfactants with $n = 3$, $n = 6$, and $n = 11$, and alkyl ethoxylate surfactants with $n = 6$ and $m = 11$ (see Figure 3.1). The simulations are labeled T3, T6, T11, and CE, respectively. This choice of surfactants covers trisiloxane surfactants that are too long, too short, and just right for superspreading [95] and one additional surfactant for reference. Starting structures for the droplets were generated by creating a cylinder from equilibrated bulk configurations of water. Surfactants were added to the surface to generate a fully covered droplet. Because trisiloxane surfactants are soluble, surfactants were added to the interior of the droplet to have a surfactant mass fraction $w_S \approx 0.02$ in the droplet interior in trisiloxane simulations. Afterwards, simulations of the surfactant-laden droplet were equilibrated for 1 ns in the NVT ensemble at 300 K. To obtain fully covered droplets, we generated starting structures with different numbers of surfactant molecules at the interfaces. After equilibration, the shape of the droplets was examined. Droplets with overcrowded surfaces could easily be identified by the strong deviation from a circular droplet shape at the end of the equilibration period. Droplets with maximum surfactant concentration at the surface with only small deformations of the droplet were used in the spreading simulations.

Equilibration simulations were performed with Nosé-Hoover thermostats and barostats [68, 76, 149]. The damping factors were set to $\tau_T = 100\,\mathrm{fs}$ for the thermostat and $\tau_p = 1000\,\mathrm{fs}$ for the barostat. Integration was performed with the multiple-timestep rRESPA algorithm [148]. We used a four-stage rRESPA with a factor of two difference in the frequency of force evaluations between adjacent levels. Bonded interactions were evaluated in the innermost stage with a timestep of 0.5 fs. On the second level, we evaluated pair interactions within a distance of 6.0 Å, with interactions smoothly shifted to zero beginning at 4.5 Å. On the third level, pair interactions up to a distance of 10.0 Å were computed, with the potential shifted to zero starting at 8.0 Å. Long-ranged interactions were computed on the outermost level.

The initial configurations for the spreading simulations were assembled from the pre-equilibrated polymer slabs and droplets. The starting configurations for the spreading simulations are depicted in Figure 3.10 for the different surfactants. The droplet radius for the T11 simulation was chosen to be larger because the surfactant was larger. We also note that although the droplet diameter for this surfactant was larger, the number of molecules that can be placed on the interfaces was much lower for this surfactant because of the overlap of the hydrophilic tails resulting from the strong curvature of the nanoscale droplet. Further details on the simulation setup is given in Table 3.4.

system	n_{pol}[a]	n_{wat}[b]	$n_{\text{surf,bulk}}$[c]	$n_{\text{surf,int}}$[d]	A_{mol}[e]	radius[f]	L_x[g]	L_y[h]	L_z[i]
					[Å^2]	[Å]	[nm]	[nm]	[nm]
PP, T6	300	55569	52	800	70	135	50	8	40
PP, T3	300	78280	83	950	60	135	50	8	40
PP, T11	360	80665	50	504	115	170	60	8	50
PP, CE	300	59309	0	1078	50	135	50	8	40
PTFE, T6	310	55569	52	800	70	135	50	8	40
PEO, T6	340	55569	52	800	70	135	50	8	40

[a] number of polymer molecules; [b] number of water molecules; [c] number of surfactant molecules in the bulk; [d] number of surfactant molecules at the interfaces; [e] approximate surface area per surfactant molecule at the interface; [f] approximate initial radius of the droplets; [g,h,i] box dimensions in x, y, and z dimension;

Table 3.4: Setup for surfactant enhanced spreading simulations.

The choice of thermostat can influence the simulation results in non-equilibrium simulations such as the spreading simulations reported here. Specifically, Heine et al. [191] showed that spreading velocities can be effected by the thermostat, which is why we performed our simulations with two different thermostatting strategies. In the first approach, labeled LN below, the lower 5 Å of the substrate were held rigid, while the region from 5 Å to 10 Å was coupled to a Langevin thermostat [167] to serve as an energy sink. The rest of the droplet was integrated using Newtonian dynamics. In a second approach, labeled NH below, the lower 5 Å of the substrate were also held rigid. The rest of the droplet was very weakly coupled to a Nosé-Hoover thermostat with damping factor $\tau_T = 10000\,\text{fs}$. Because of the relatively large system size with $\mathcal{O}(3 \times 10^5)$ particles and the large coupling constant τ_T, the thermostat has a very weak effect on the dynamics [76].

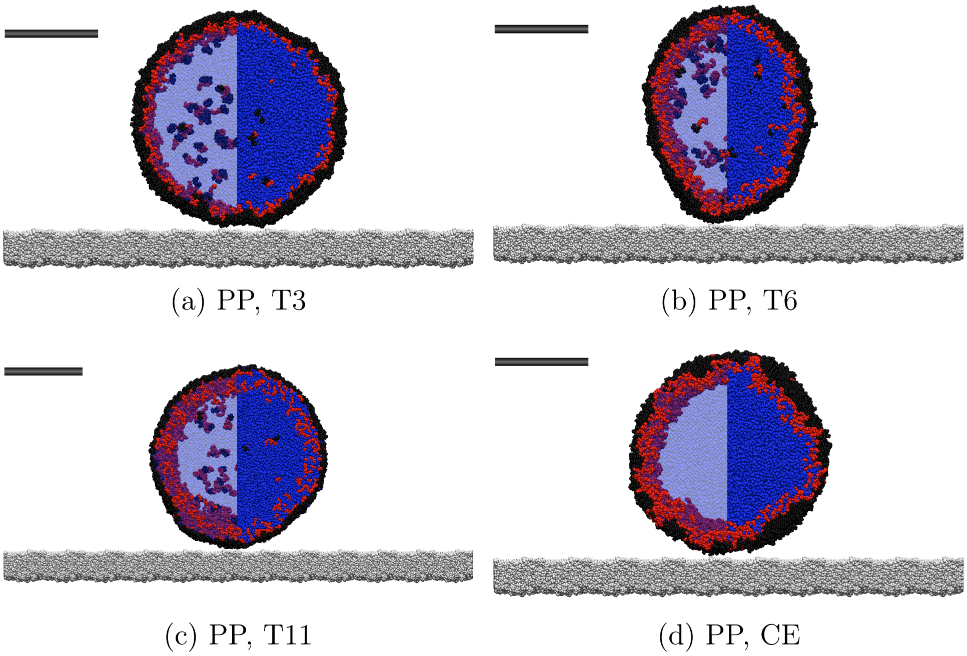


Figure 3.10: Snapshots of the starting configurations in simulations with (a) T3, (b) T6, (c) T11, and (d) CE on a PP substrate. Gray: substrate; blue: water; red and black: hydrophilic and hydrophobic part of the surfactant. The water in the left half of each droplet is transparent. The black bar in each figure has a length of 10 nm. (Reprinted with permission from Ref. [190])

We used the PPPM algorithm to compute both long-range electrostatics [34] and dispersion interactions in all simulations. Settings for the PPPM method were as suggested in Section 2.8. In spreading simulations, we used a two-stage rRESPA, with a factor of two between the inner and outer timesteps. The inner timestep, on which bonded an pair interactions up to 7.0 Å were computed, was 1 fs. Nonbonded interactions on the inner level were shifted to zero between 6.5 Å and 7.0 Å. Long-ranged interactions were computed on the outer level.

3.3.1.2 Droplet geometry and spreading exponents

The base width of a droplet is the width of the solid–liquid contact area. Although the lower area of the droplet is not a circle in our simulations, we will in the following use the commonly used term “base radius” r for half of the base width. For the droplet simulations reported here, several challenges need to be overcome to compute the base radius. First, as the polymer substrate is flexible and amorphous, its surface is not as planar as those of crystalline substrates. Hence, a definition of the exact position z_{base} of the surface is not obvious. Second, if there are precursors or feet close to the three-phase contact line, the shape of droplets can deviate from their macroscopic spherical or cylindrical cap shapes [10]. In an MD simulation, which can resolve such deviations in the droplet shape, the base radius of the droplet right at the substrate surface is thus not necessarily comparable to the value reported in experiments.

The base radius r is computed in two steps. In the first step we define the position of the solid–liquid interface z_{base}. A histogram of the density of the water and surfactant molecules is computed with 1 Å-wide bins in the x and z dimensions using a procedure similar to that of [144]. The densities are averaged over 10 ps, a time scale long enough to obtain data that is not too noisy, but also short enough to avoid distortion of the data because of changes in droplet shape that occur on much larger time scales. From the density histogram, the droplet domain is determined. Because the vapor phase is almost empty and the interface has a steep density gradient, the domain can be simply defined according to the density: any bin whose density is larger than 10 percent of the maximum density is considered "inside" the droplet. Bins considered to be inside

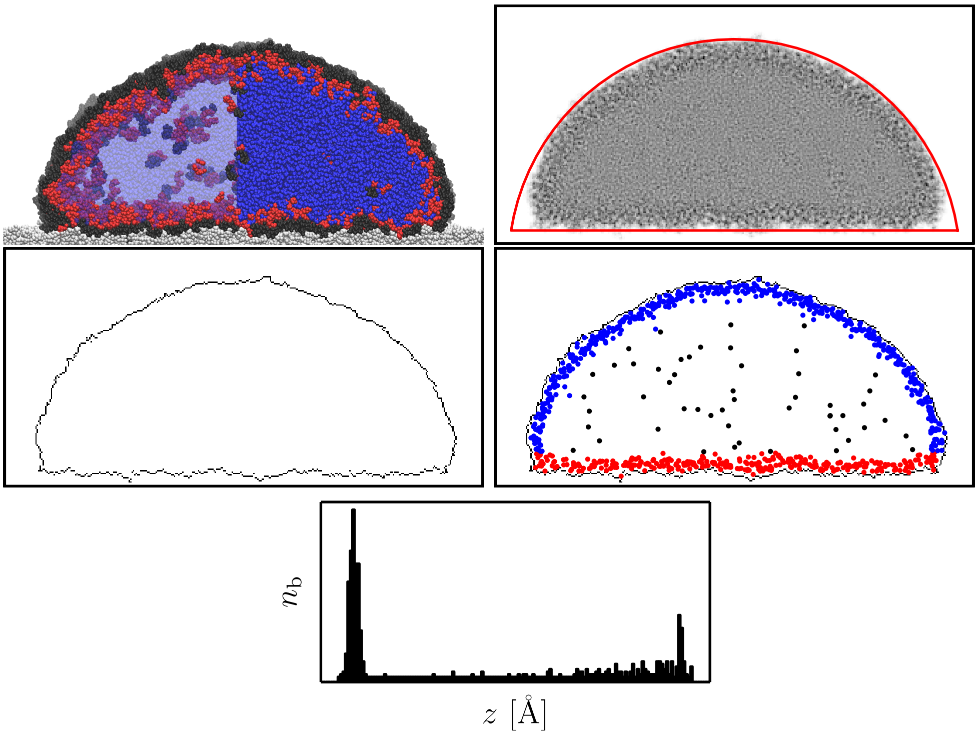


Figure 3.11: Analysis steps. Upper left: Simulation snapshot. Upper right: Density, with a circular fit defined from the boundary elements, and a horizontal line at z_{base}. Middle left: Boundary elements. Middle right: position of the center surfactant atoms given in Figure 3.1. The atoms are labeled according to whether they are classified as at the liquid surface (blue), solid–liquid interface (red), or in the bulk (black). Bottom: Histogram of the z position of the boundary elements. z_{base} is the z position of the highest peak. (Modified reprint with permission from Ref. [190])

the droplet but next to a vapor element are considered boundary elements. Next, we compute a histogram of the z position of the boundary elements. For small values of z, the histogram has a peak which results from the flat surface close to the substrate. The position of the peak is used as the base position z_{base} of the droplet. The procedure of defining z_{base} is illustrated in Figure 3.11.

In the second step we use a method similar to that suggested by Halverson et al. [115] to compute the base radius r of the droplet. As base radius, we use half the width of a rectangular cuboid that contains a fraction p of all atoms within a distance h from the base position z_{base} of the droplet. We tried different values for h and p and found that results, especially for the spreading exponent, are insensitive to these parameters. In the following, we report results for $h = 10\,\text{Å}$ and $p = 0.99$.

Different spreading regimes were determined from log-log plots of the base radius over time. The spreading exponents were determined from fitting straight lines to the linear regimes of the log-log data.

The height h_d of the droplet is defined as the difference between the highest boundary element and the base position of the droplet. Using the approximation that the droplet has the shape of a cylindrical cap, the contact angle θ can be determined from

$$h_d = r\frac{1 - \cos(\theta)}{\sin(\theta)}. \tag{3.13}$$

To examine the macroscopic droplet shape we perform a circular fit to the droplet boundary. In the fit, we only include points of the boundary farther than 50 Å from the base position z_{base}. This still leaves a sufficiently large fraction of the droplet to be included in the fit while eliminating the influence of the base region of the droplet. This choice still reasonably represents the macroscopic part of the droplet, as shown in Figure 3.11.

3.3.1.3 Surfactant concentration at the interfaces

Surfactants are considered as adsorbed at the interfaces if their central atom (defined in Figure 3.1) is within 15 Å of the droplet boundary. Among these surfactants, those whose central atom is within 15 Å of the substrate are classified as adsorbed to the solid–liquid interface, whereas the others are classified as adsorbed to the liquid–vapor interface. All other surfactants are considered as not adsorbed but in the interior of the droplet. The value of 15 Å was chosen because it gives a reasonable separation of the different regions, as shown in Figure 3.11. The total surface area of the droplet is approximated by the number of border elements. The size of the bottom area is approximated by the number of border elements of the droplet less than 15 Å above the base position; the rest of the droplet surface is classified as as part of the liquid–vapor interface. The surfactant concentration at the different interfaces is the ratio of the number of surfactants at an interface divided by the size of the interface.

3.3.1.4 Moment-based surface analysis of the droplet shape

Differences in the droplet shape at the contact line were quantified using methods from image analysis with a two step approach. In the first step, we use the binary Mumford-Shah segmentation model [192] to define the location of the interface of the droplet. In a second step, we apply the moment-based analysis algorithm for implicit surfaces by Berkels et al. [193] to locally classify the surface of the droplets. The entire process is illustrated in Figure 3.12.

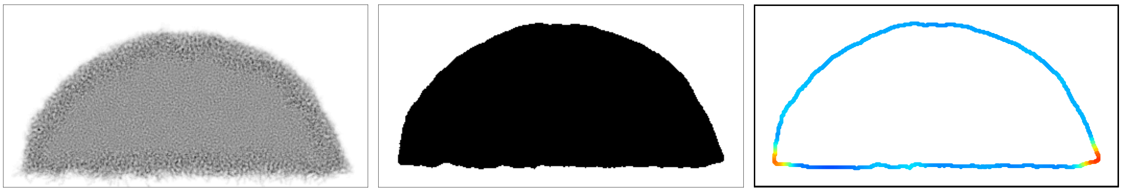


Figure 3.12: Steps to classify the local shape of the droplet. Left: input image for the Mumford-Shah segmentation; middle: separated condensed and vapor domains; right: droplet surface, where the color code denotes the value of the surface classifier $C(x)$. The surface classifier identifies sharp transitions at both leading edges of the droplet. (Reprinted with permission from Ref. [190])

We here use the binary Mumford-Shah segmentation functional

$$E(\rho_c, \rho_v, \Omega_c) = \int_{\Omega_c} (\rho - \rho_c)^2 \mathrm{d}A + \int_{\Omega \setminus \Omega_c} (\rho - \rho_v)^2 \mathrm{d}A + \eta \mathrm{Per}(\Omega_c), \tag{3.14}$$

where ρ is the histogram of the local densities of the droplet (cf. Section 3.3.1.2), ρ_c and ρ_v are the constant densities of the condensed and vapor phase, Ω is the computational domain, Ω_c and $\Omega \setminus \Omega_c$ are the domains of the condensed and vapor phase, and $\mathrm{Per}(\Omega_c)$ is the perimeter of Ω_c in Ω, which is essentially the length of the interface between the condensed and vapor phase. The interface of the system is defined by minimizing Equation (3.14) with respect to ρ_c, ρ_v, and Ω_c. The last term in the functional is a penalty term; η is a parameter that controls the weight of this term and can be adjusted to control the smoothness of the generated segmentation. A large value for η will decrease the interface by smoothening edges. We used $\eta = 0.005\,\mathrm{g\,Å/cm^3}$ in our computations because it provided relatively smooth droplet shapes while preserving the shape of the droplet well. For the minimization with respect to the set Ω_c, the unconstrained convex reformulation proposed in [194] is used. The resulting convex minimization problem is solved using a first-order primal-dual algorithm [195]. For fixed ρ_c and ρ_l, this allows us to find a global minimizer of the non-convex binary segmentation problem. The segmentation was performed with the Quocmesh library [196].

The contour of the computed condensed phase is classified using moment-based surface analysis. The zero-moment shift of the boundary of the object Ω_c is defined as $M^0_\epsilon[\Omega_c](x)$

$$M^0_\epsilon[\Omega_c](x) = \frac{1}{\pi\epsilon^2}\int_{B_\epsilon(x)} d_{\Omega_c}(y)(y-x)\mathrm{d}y, \tag{3.15}$$

where x is a point on the boundary of Ω_c, the integral is performed over a disk $B_\epsilon(x)$ centered at position x with radius ϵ, and $d_{\Omega_c}(y)$ is the signed distance function of Ω_c. The distance function gives the Euclidean distance of any point in the image to the boundary of the object. The signed distance function is equal to the distance function with the difference that it is defined as positive outside the object and negative inside the object. The surface classifier $C(x)$ is defined as

$$C(x) = g_\beta(\|M^0_\epsilon[\Omega_c](x)\|/\epsilon^2), \tag{3.16}$$

from the surface moment, where

$$g_\beta(t) = \frac{1}{1+\beta t^2}. \tag{3.17}$$

The smaller the value of $C(x)$, the flatter is the region close to position x of the surface, $C(x)$ can thus be used to characterized the surface locally. It should be noted that g_β is a monotonic function that maps the values of $\|M^0_\epsilon[\Omega_c](x)\|/\epsilon^2$ into the interval $(0,1]$. The use of this function is motivated in [193]. The function and the included parameter β are only used for scaling purpose and do not qualitatively change the results. The only parameter that influences the results qualitatively is the value of the radius of the disk ϵ, which defines the order of magnitude on which the surface should be characterized. For small values of ϵ, the shape is examined on a small scale and the result is sensitive to noise. For large values of ϵ, the classifier provides information on the large-scale contour of the shape. In our analysis we use $\epsilon = 20$ Å because this is approximately the size of the magnitude of the effect that is observed. The value of β in g_β is set to 20.

An example of the moment-based surface analysis and important features of the resulting classifier are briefly visualized in Figure 3.13. Based on the surface classifier, it is possible to discriminate edges and smooth regions. The regions at the lower left and at the upper center are identified as edges. The smooth region at the lower right can be discriminated from the sharp edges. Another important feature of the classifier is that the difference between small angles, such as in the lower left of the image, and wider angles, such as in the upper center, is recognized. The latter effect has important implications for the application of this method to our results because the value of the classifier is influenced by the value of the contact angle θ of the droplet.

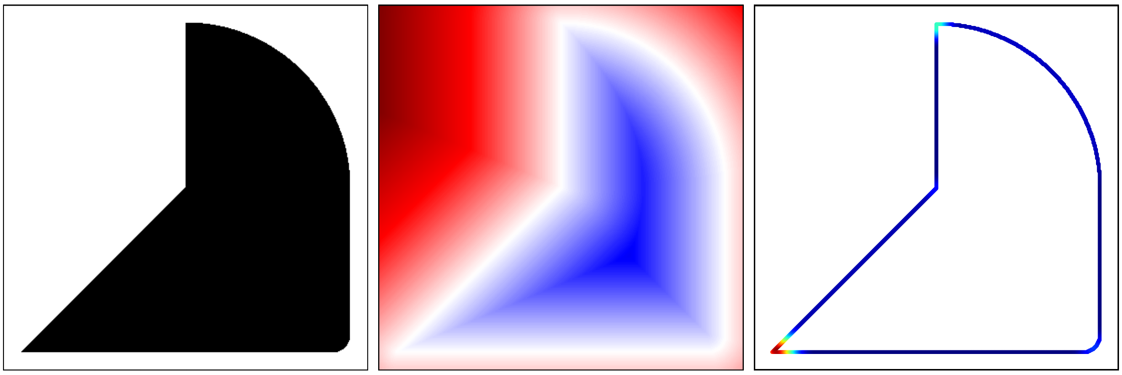


Figure 3.13: Example of the moment-based surface analysis. Left: Shape for the analysis; center: signed distance function in the image domain; right: surface of the shape color-coded with the value of the surface classifier $C(x)$. (Reprinted with permission from Ref. [190])

3.3.2 Wetting dynamics and droplet shape

3.3.2.1 Spreading regimes and exponents

We first comment on the absolute value of the base radius over time. At first glance, the upper part of Figure 3.14, which shows the base radius r of the droplet as a function of the time t in a log-log representation of the LN simulations, suggests that the droplet covered with alkyl ethoxylate surfactants spreads more rapidly than the droplet laden with the superspreading agent, because the value of the base radius r is larger at any given time. However, the base radius depends on several quantities, such as the total volume of the droplet (Equation (3.1) is just a proportionality condition). Moreover, deviations from a perfect circular droplet shape in the starting configuration, as seen in Figure 3.10, can distort the spreading velocities in the inertial regime. Consequently, the base radius of the alkyl ethoxylate-laden droplet is larger than the base radius of the trisiloxane-laden droplet. In contrast to the base radius, the spreading exponent α in the viscous regime is not influenced by these effects but arises from the underlying physics. It is thus the quantity that best describes the spreading. That α is larger for the simulation with the superspreader shows that the droplet spread faster than in the non-superspreading simulations. The spreading exponents observed in the different simulations and the time intervals of the different spreading regimes are given in Table 3.5.

		initial regime		viscous regime		
thermostat	system	t [ns]	α	t [ns]	α	superspreading[a]
LN	PP, T6	0.2 to 2.4	0.39	2.4 to 5.2	0.27	yes
	PP, CE	0.2 to 2.0	0.34	2.0 to 5.2	0.20	no
NH	PP, T6	0.2 to 2.0	0.34	2.0 to 8.0[b]	0.20	yes
	PP, T3	0.2 to 1.8	0.29	1.8 to 8.47[b]	0.17	no
	PP, T11	0.2 to 2.0	0.25	2.0 to 4.2	0.17	no
	PP, CE	0.2 to 2.0	0.28	2.0 to 9.0[b]	0.14	no
	PTFE, T6	0.3 to 3.6	0.37	3.6 to 10.0[b]	0.16	no
	PEO, T6	0.2 to 2.8	0.37	2.8 to 7.7[b]	0.21	no

[a] "Yes" indicates superspreading is observed in experiments at these conditions. [b] end of reported viscous regime corresponds to end of the simulation.

Table 3.5: Spreading exponents and duration of different spreading regimes. (Modified reprint with permission from Ref. [190])

Similar to experimental observations, the spreading passes through several regimes. The first regime is characterized by a high spreading exponent. While the spreading exponent that we observe in this regime is similar to the inertial regime in macroscopic experiments [99] the transition time between the first and the second regime is moved to much larger values of the characteristic inertial time $\tau_c = (\rho R^3/\gamma)^{1/2}$, where ρ is the liquid density, R is the initial radius of the droplet, and γ is the surface tension of the liquid. For macroscopic droplets, the transition time is $\tau = \mathcal{O}(\tau_c)$ [99], whereas in our simulations the transition time is $\tau = \mathcal{O}(100\tau_c)$, with $\tau_c = \mathcal{O}(10\,\mathrm{ps})$. This large difference is potentially caused by the small length scales in our simulations compared to experiments, or could result from using cylindrical droplets instead of the spherical droplets used in experiments.

The second regime is the viscous regime, characterized by a lower spreading exponent. For this second regime, linear fits used to determine the spreading exponent are shown in Figure 3.14. The measured data are approximately on a straight line in this regime. At later times in the LN simulations, the spreading slows down and the curves level off. In this third regime the surfactant concentration at the interfaces decreases. Moreover, because of the thermostatting strategy, the temperature strongly increases in LN simulations. As a result, the driving force for spreading weakens. Representative snapshots of the simulations for the different regimes are depicted in Figure 3.15.

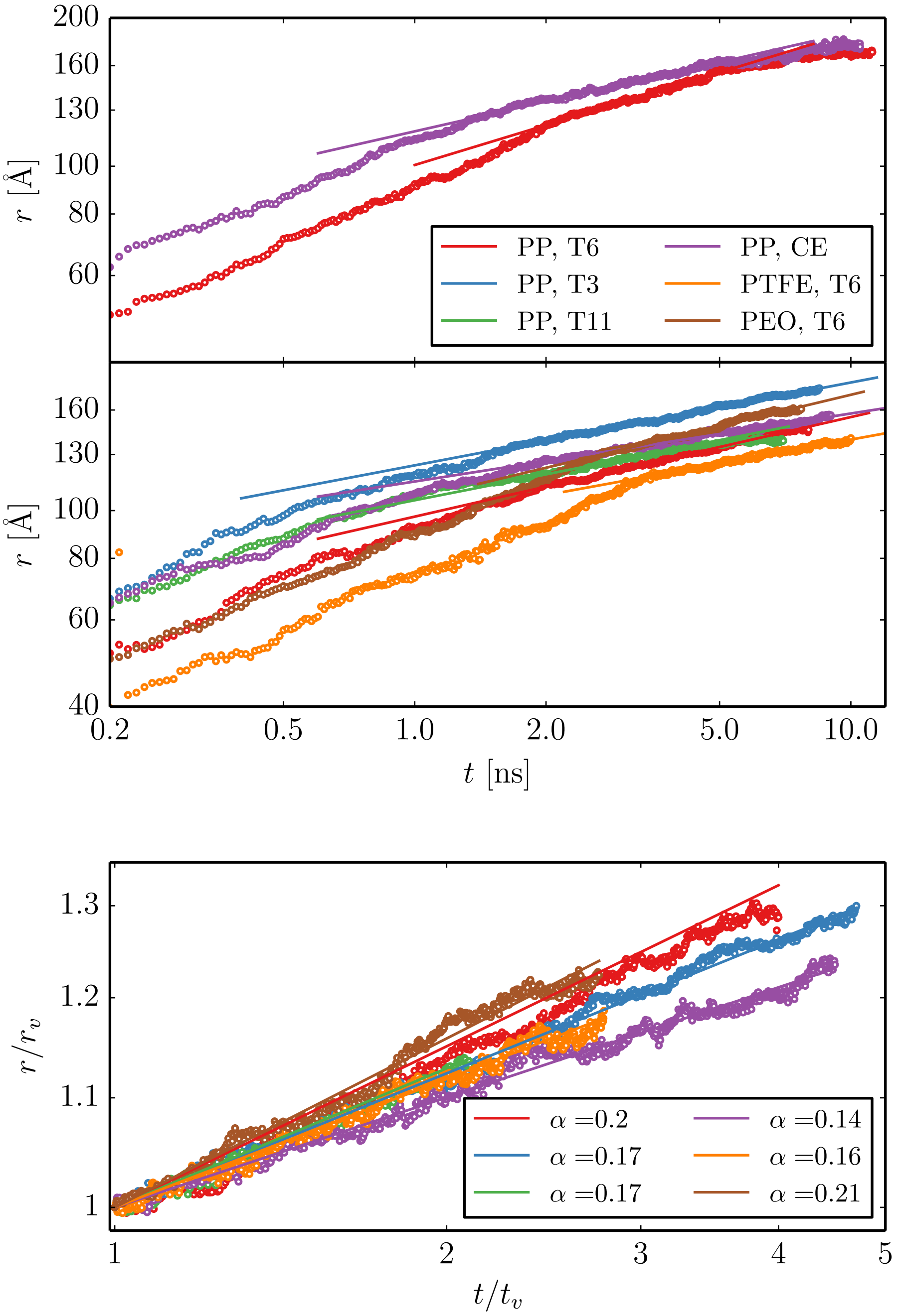
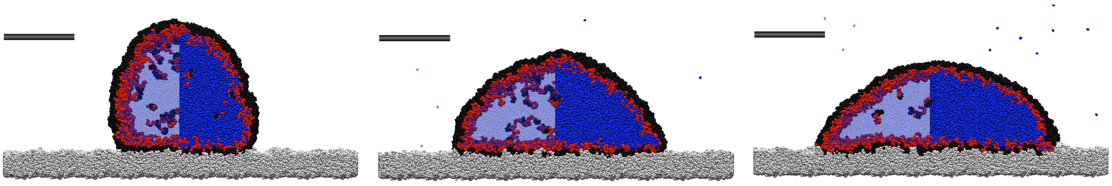


Figure 3.14: Base radii r as a function of the simulation time t in log-log representation and lines to determine the spreading exponent. Top: LN simulations; middle: NH simulations; bottom: results from the viscous regime from the NH simulations shifted such that differences between the lines are highlighted. (Modified reprint with permission from Ref. [190])



(a) $t = 1\,\text{ns}$, inertial regime (b) $t = 3.7\,\text{ns}$, viscous regime (c) $t = 10.0\,\text{ns}$, final regime

Figure 3.15: Snapshots of LN simulation with T6 surfactant on PP. Colors are as in Figure 3.10. In the inertial regime, dynamics are influenced by the initial droplet shape. In the viscous regime, the initial shape information is lost and viscous dissipation controls the spreading velocity. In the third regime, the droplet surface area increases greatly and the surfactant density at the interfaces and inside the droplet is reduced.

The third regime is a result of the simulations being performed at the nanoscale; a connection to the macroscale is difficult and further investigation of this regime is unlikely to enhance understanding of superspreading. For the inertial regime, no relevant differences were found between superspreading and non-superspreading surfactants in experiment [101]. For the second, viscous regime, however, significant differences between superspreading and non-superspreading solutions were observed [101]. The behavior of droplets and especially the spreading exponents in this regime are thus of interest.

For the viscous regime, we find that the spreading exponent for the superspreading surfactant is greater than the spreading exponent for the non-superspreading solution in LN simulations. However, comparing the spreading exponents observed by Wang et al. [101] with ours, and considering the different geometries, the spreading exponents in the LN simulations appear to be too large: Wang et al. observed spreading exponents $\alpha \approx 1/7$ for superspreading and $\alpha \approx 1/10$ for non-superspreading solutions, which corresponds to the MKT and the HDM. We therefore would expect to find exponents of $\alpha \approx 1/5$ and $\alpha \approx 1/7$. However, the observed spreading exponents are larger still (cf. Table 3.5).

When examining the temperature of the LN simulations, we noticed that over the time of the simulation the temperature increased by more than 100 K. Apparently the heat sink in the Langevin region was not sufficient to adsorb the heat generated by dissipation, or the heat could not be transported sufficiently fast to the heat sink. This motivated the NH simulations, in which the majority of the particles is weakly coupled to a Nosé-Hoover thermostat. In these simulations the temperature did not increase during the simulation despite the weak coupling. Because we did not expect to recover the superspreading regime in our simulations, NH simulations were stopped when the viscous regime was sufficiently well-developed to determine spreading exponents. The base radius as a function of time is given as log-log data in the middle image of Figure 3.14; the lower image of Figure 3.14 shows the data of the viscous regime translated to highlight the differences between the curves. Lines from the fit to determine the spreading exponent are also shown.

For the NH simulations, results are in much better agreement with experimental data [101]. The spreading exponent is smallest for the alkyl ethoxylate surfactant ($\alpha \approx 0.14$); for trisiloxane surfactants in conditions that are too long or too short and where the substrate is too hydrophobic for superspreading, the spreading exponent is $\alpha \approx 0.17$. Considering the inherent noise, the results for these non-superspreading scenarios are in reasonable agreement with the HDM, such as the experiments by Wang et al. [101] with non-superspreading solutions of trisiloxanes on PP substrates. The simulated values are therefore in reasonable agreeent with those from experiment, especially when considering that also in experiment slight deviations from the ideal value of $\alpha = 1/10$ were observed. For the superspreading case, with the T6 surfactant on the PP substrate, the measured spreading exponent is slightly above 0.2, which is in reasonable agreement with MKT and therefore also in agreement with the experimental results [101]. For the PEO substrate, for which no experimental data is available, the spreading exponent is also slightly above 0.2.

As discussed above, differences in spreading exponents suggest that different physical processes underlie the spreading. That the experimentally observed difference between superspreading and non-superspreading solutions is recovered in our simulations suggests that effects relevant to the superspreading mechanism have been captured. The reason for the increased spreading exponent for the superspreading case will be discussed in Section 3.3.2.2. The differences in the physical behavior for the PEO simulation compared to the other simulations is addressed here: As described above, the PEO substrate is hydrophilic and water has an affinity to this substrate. A result of this affinity is that in our simulations, water diffused to the substrate at the contact line and surfactants desorbed from the substrate, as shown in Figures 3.16 and 3.17. This desorption of surfactants or even depletion of surfactants close to the contact line was not observed in any other simulation, which can be seen from the surfactant concentration at the substrates decreasing much less rapidly for all other simulations compared to the simulation with the PEO substrate. We therefore suggest that the increased spreading exponent is a result of water diffusing to the contact line region and pushing surfactants aside. It is unclear, however, how exactly this process increases the spreading exponent. We would also like to mention that it is unlikely that this effect can be captured in an experimental setup because the time scales of surfactant depletion at the contact line region is on the order of magnitude of a few nanoseconds, i.e. the time scale of the simulations reported here. This time scale cannot be resolved in experiment of spreading in the lab.

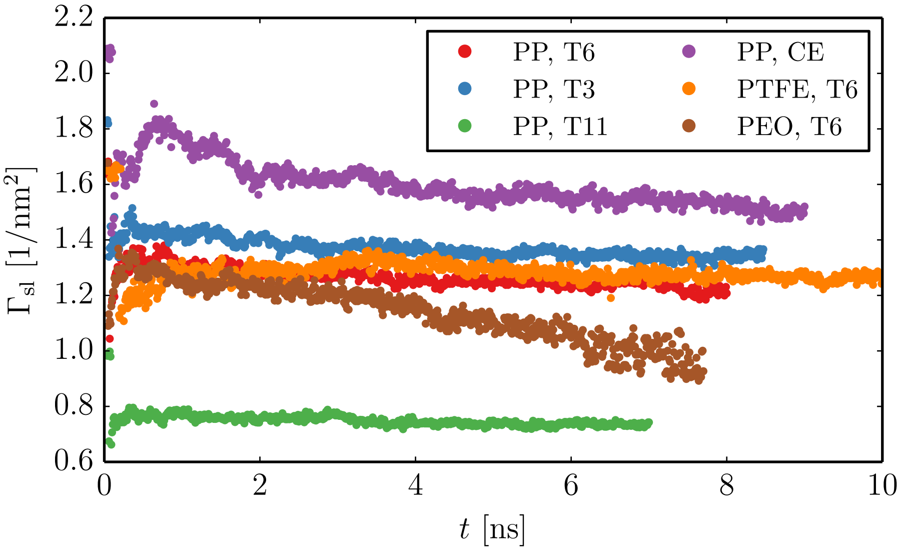


Figure 3.16: Surfactant concentration Γ_{sl} at the solid–liquid interface for the different NH simulations. For the simulation on PEO, the surfactant concentration drops more rapidly because of unfavorable surfactant adsorption on the substrates compared to water adsorption. (Modified reprint with permission from Ref. [190])

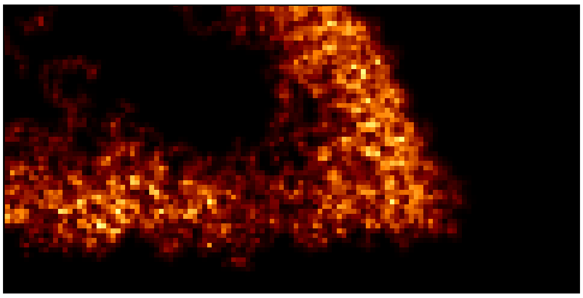
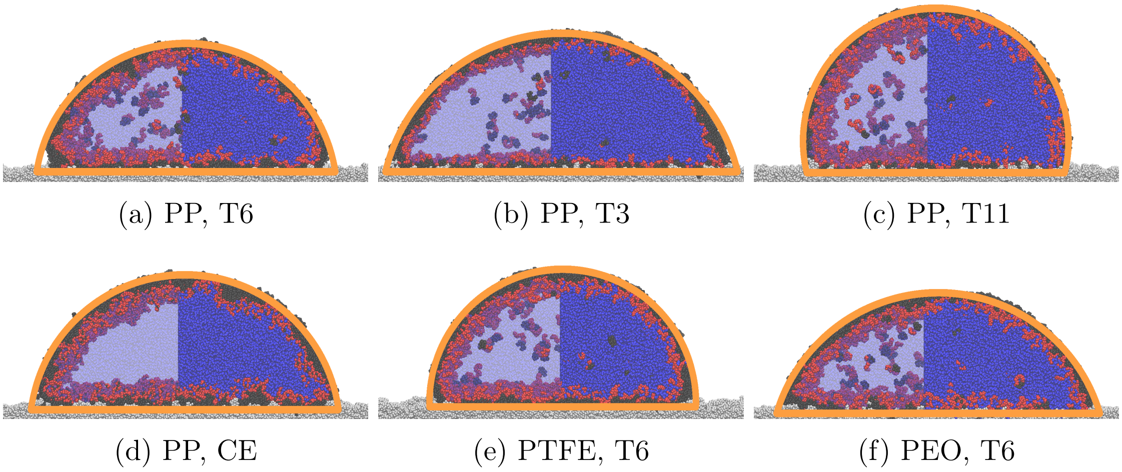


Figure 3.17: Surfactant density close to the three-phase contact line for the PEO simulation. The region at the substrate close to the contact line is depleted from surfactants.

3.3.2.2 Droplet shape

As outlined in Section 3.1, a molecular understanding of phenomena at the contact line region is of great interest to better understand superspreading. Snapshots of the droplets of the different NH simulations with the circular fit to the droplet shape are depicted in Figure 3.18. MD simulations with simple model molecules by Shen et al. [114] suggested that bilayers might form in spreading droplets with trisiloxane surfactants. In our larger, more realistic simulations, however, it is immediately apparent that neither a bilayer nor a precursor forms. Likewise, feet are not visible, which argues against theories based on precursor formation. The results in this Figure also show that the approximation of the droplet shapes with cylindrical caps is reasonable.


(a) PP, T6

(b) PP, T3

(c) PP, T11

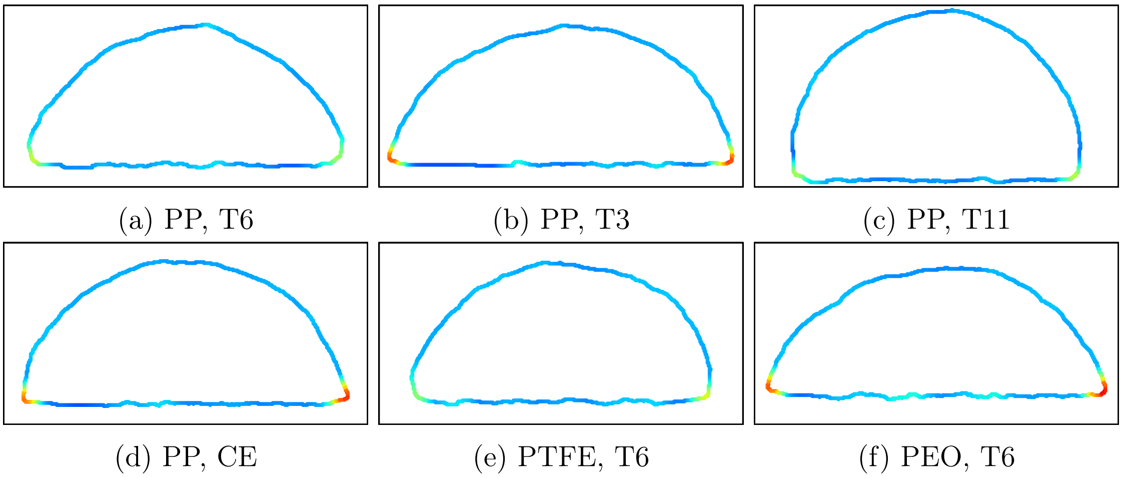
(d) PP, CE

(e) PTFE, T6

(f) PEO, T6

Figure 3.18: Simulation snapshots from the NH simulations in the viscous regime. The orange lines are the circular fit and a horizontal line at the base position z_{base}. No bilayer, precursor, or foot is formed in any of the simulations. (Reprinted with permission from Ref. [190])

Results for the moment-based surface analysis are depicted in Figure 3.19 and 3.20. The first of the figures shows the surface of the final snapshot of the viscous regime of each simulation color-coded with the edge classifier. The latter figure shows the maximum value of the edge classifier $C_{\max}$ in the left and right half of the droplet (which is essentially at the leading edge, as can be seen from Figure 3.19) over the droplet contact angle for all snapshots from the viscous regime of the NH simulations. In the subfigures in Figure 3.20, the results obtained with the superspreading scenario are given in every of the subfigures such that a comparison between the superspreading and non-superspreading cases is possible. For the results in Figure 3.20 the time is implicity included in the figures. Because the droplets spread over time the contact angle gets smaller with increasing simulation time. With increasing simulation time, the results thus move from right to left in the images. In addition, with decreasing contact angle, the value of the surfactant classifier tends to grow, as the macroscopic contact angle has an impact on the moment-based analysis, as briefly mentioned in Section 3.3.1.4.



(a) PP, T6

(b) PP, T3

(c) PP, T11

(d) PP, CE

(e) PTFE, T6

(f) PEO, T6

Figure 3.19: Droplet shapes from the NH simulations in the viscous regime color-coded with the classifier $C(x)$. (Reprinted with permission from Ref. [190])

The differences between some of the non-superspreading cases compared to the superspreading case is immediately visible from Figure 3.19. The droplet at superspreading conditions (PP, T6) has a smooth transition at both droplet edges, which can be seen from the shape but also from the surface classifier $C(x)$. For the simulations at non-superspreading conditions with the surfactant that is too short (PP, T3), the alkyl ethoxylate surfactant (PP, CE), and the surface that is too hydrophilic (PEO, T6), the leading edges of the droplet are much sharper. At conditions where the surfactant is too long for superspreading (PP, T11), the visual impression from the shape is that the transition from the vapor-liquid interface to the solid-liquid interface is sharper compared to what is observed for the droplet at superspreading conditions. However, this feature is not captured well by the image classifier, as can be seen from the color of the lines. For the case that is too hydrophobic for superspreading (PTFE, T6), a strong difference in the surface classifier compared to the superspreading case is not visible.

The differences between the droplet at superspreading conditions and the non-superspreading cases with a too-short surfactant, the alkyl ethoxylate, and the too-hydrophilic substrate can also be detected when comparing the maximum values of the surface classifier during the viscous spreading regime. As can be seen from Figure 3.20, there is a clear separation of the results obtained for the droplet shape at superspreading conditions and the three non-superspreading cases for which differences are also easily visible from Figure 3.19, i.e. for (PP, T3), (PP, CE), and (PEO, T6). It should be noted that the differences between the computed values for the classifier are numerically small; however, these small differences reflect the strong differences in

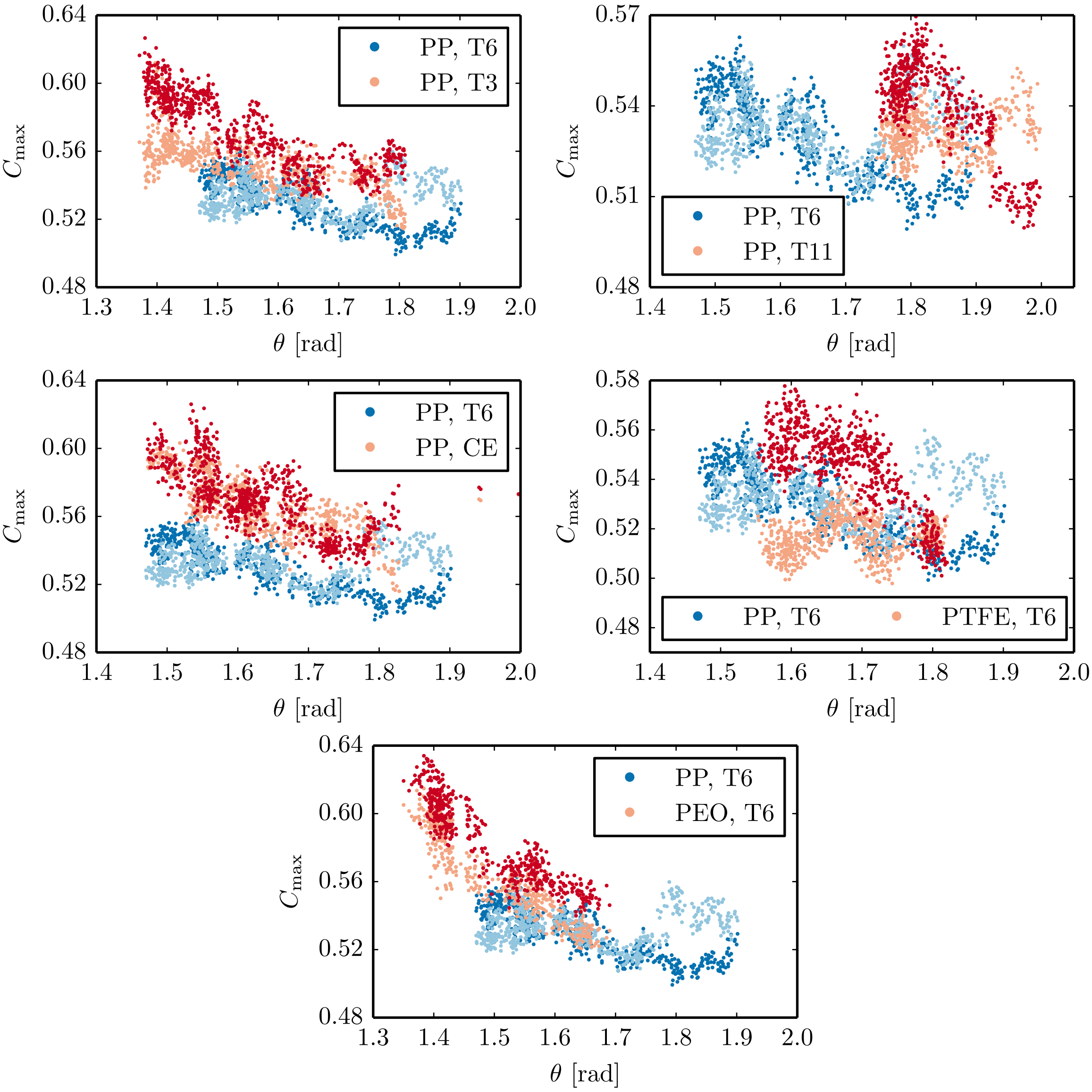


Figure 3.20: Maximum droplet classifier C_{max} of the left and right edge of the droplet over the contact angle during the viscous spreading regime. The bright and dark shades of the symbols are the results for the left and right edge of the droplet. The data for PP, T6 are given in every image to facilitate direct comparison. (Modified reprint with permission from Ref. [190])

the shapes that are visible in Figure 3.19. That the resulting numerical differences are that small is a feature of the method.

Differences in the results are also visible for the too-long surfactant. Because the contact angle was mainly sampled in different regions for this simulation and the simulation at superspreading conditions, the discrimination is less obvious. Considering, however, that the surface classifier has the tendency to grow with decrasing contact angle, the observed results suggest that a difference exists for these cases, too.

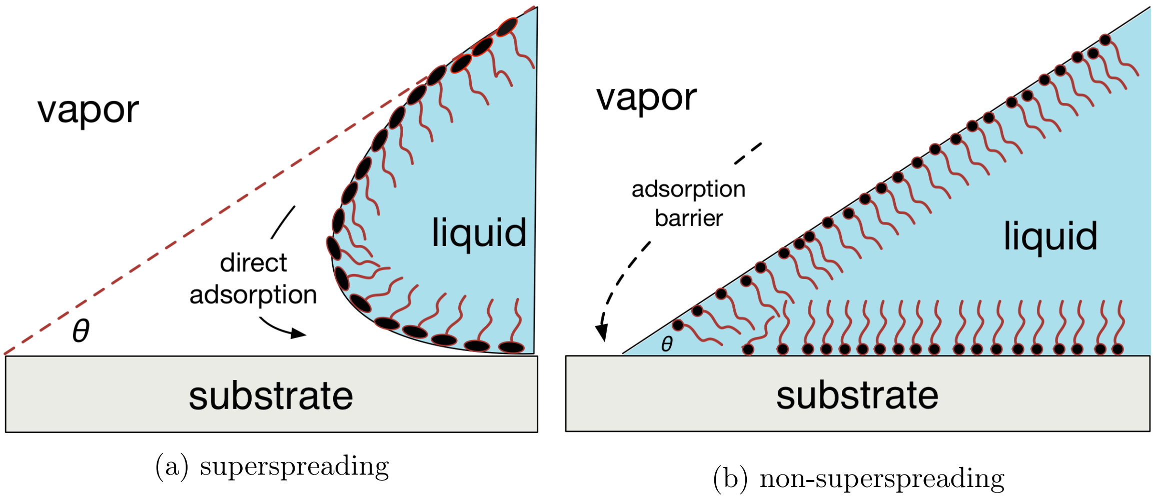
The only case where a separation of the results obtained with the superspreading case and the non-superspreading case is not possible is the scenario in which the substrate is too hydrophobic. In agreement with what is seen in Figure 3.19, one edge of this droplet seems to be sharper than what is observed at superspreading conditions, whereas the other is smoother.

3.3.3 Relevance for superspreading

Two remarkable features of the simulation at superspreading conditions are that the contact line lies behind the circular fit and that the transition from the liquid–vapor to the solid–liquid interface is smooth. This unusual mechanism, to our knowledge, has previously been observed only in the model LJ surfactant study of McNamara et al. [112] This transition removes the sharp edge at the contact line. This droplet shape is thus a mechanism which can overcome the Huh–Scriven paradox [12], which results from sharp droplet edges, and may possibly explain why the spreading exponent in the viscous regime is larger for this simulation.

As mentioned above, Karapetsas et al. [107] showed that superspreading can occur if direct surfactant adsorption through the contact line is possible and surfactants are soluble. While the latter condition is known to be fulfilled for trisiloxane surfactants, the direct adsorption mechanism remained obscure. The smooth transition from the liquid–vapor to the solid–liquid explains how direct adsorption through the contact line can work, as sketched in Figure 3.21. This smooth contact-line transition can therefore potentially be the molecular mechanism of superspreading. That simulated contact line regions do indeed resemble the proposeded mechanisms can be seen from Figure 3.22. In the following, we discuss why this smooth transition was observed in simulation at superspreading conditions and relate this to the complex dependencies of the superspreading mechanism on the surface energy of the substrate and the surfactant chain length.

The smooth transition between the interfaces requires a strong bending of the droplet surface close to the contact line. This bending is associated with an energy penalty



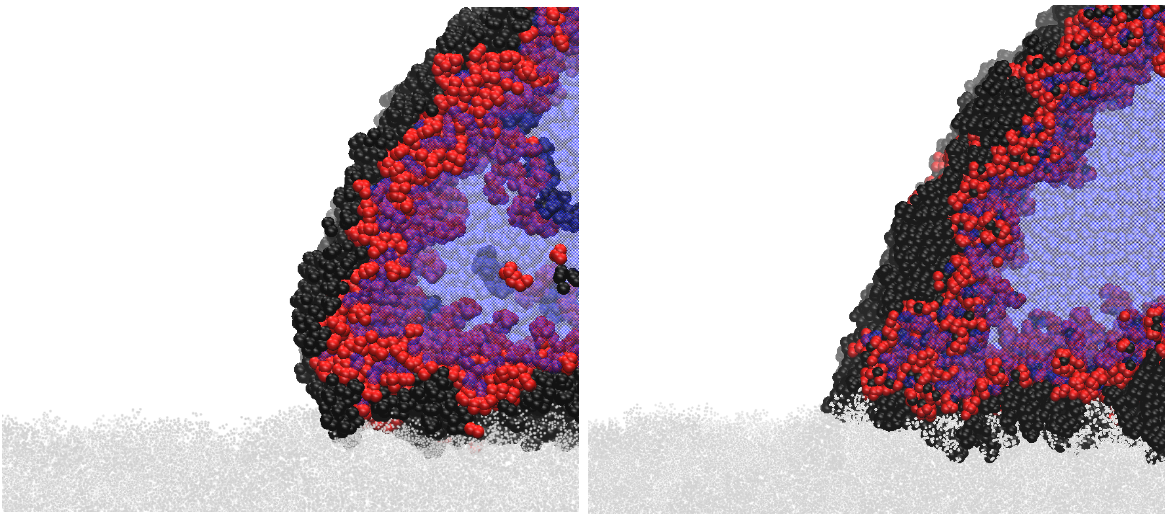
(a) superspreading

(b) non-superspreading

Figure 3.21: A sketch of the proposed mechanism for superspreading compared to usual surfactant enhanced wetting. Gray: solid; blue: water; black and red: hydrophilic and hydrophobic parts of the surfactants. The angle θ is the macroscopic contact angle of the droplet. For the superspreading case, the contact line is bent inwards, allowing for direct surfactant adsorption on the substrate. The hydrophilic tails are long enough to form aggregates, but not so long as to overlap and repel each other in the bended region. The compact head group and the relative size of the head and tail groups enhance the stability of these aggregates. The preferable adsorption energies and that aggregates are not torn apart compensate for the bending energy penalty. For conventional surfactant-enhanced spreading, the surfactant aggregates are torn apart: surfactants must reorient at the contact line, thus imposing an adsorption barrier. (Reprinted with permission from Ref. [190])

and will thus only occur if this energy penalty is compensated for by an energetically favorable feature. We hypothesize that this feature is the existence of an unbroken surfactant aggregate at the interfaces; the existence of these aggregates for superspreading trisiloxane surfactants at water surfaces was shown by Ritacco et al. [197]. That the smooth transition only occurs for the superspreading scenario at intermediate values of the chain length of the surfactant is a logical consequence of this hypothesis.

The chain length of the surfactants has an impact on the energy penalty associated with the bending of the interface and the stability of the aggregates. For surfactants that are too long, the hydrophilic parts will overlap and repel each other when the interface is bent, so this state will be energetically unfavorable and surfactant aggregates will break up. For surfactants that are too short, surfactant aggregates at the interface are not formed, as shown experimentally in [197]. Thus, intermediate values for the chain length are best to facilitate the smooth transition at the contact line.



(a) superspreading

(b) non-superspreading

Figure 3.22: Snapshot of the contact line region for the superspreading (left) and non-superspreading PP, CE (right) simulations. The geometries resemble the proposed mechanisms in Figure 3.21. (Reprinted with permission from Ref. [190])

For surfaces that are too hydrophilic, like PEO, the adsorption of the surfactant is unfavorable because the surface prefers contact with water. This can be seen from the surfactant concentration at the solid–liquid interface in Figure 3.16, where it is shown that the surfactant concentration on the PEO substrate drops more rapidly than on the other substrates, indicating that surfactants desorb from the interface or are repelled by water close to the contact line, which will break the surfactant aggregates, which is visible from Figure 3.17.

For the simulations with a substrate that is too hydrophobic, it seem from our simulations that the transition from the liquid-vapor interface is smooth. This suggests, that the fast adsorption through the contact line is in principle possible. That the path may exist, however, does not necessarily mean the the surfactants will take this path to adsorb rapidly on the substrate: Substrates with low surface energies, such as PTFE, have weak interactions not only with water, but also with the surfactants. Adsorption on these surfaces is thus not very favorable. A possible explanation could also be that our model is not sufficiently accurate for realistic simulations on that substrate.

The contact-line mechanism at superspreading conditions observed in this study does not explain why spreading rates are greatest for intermediate surfactant concentrations, or more precisely, why spreading rates are maximal right before a phase transition from vesicles to lamellar phases occurs inside the droplet. To show that this is not an arguement against the proposed molecular mechanism, this feature is addessed briefly: The maximum spreading rates below the phase transition are unrelated to the mechanism at the contact line, but to how fast surfactants can be transported from the bulk to the interfaces. In the solution with vesicles, surfactant transport will be faster with greater surfactant concentration; however, when surfactants are in lamellar phases, the surfactants must pass through a phase transition before they can adsorb to the interfaces, which is why surfactant transport to the interfaces is slower above the phase transition, which is a plausible explanation for the role of vesicles and maximum spreading rates as a function of the surfactant concentration.

Before concluding this chapter, we briefly discuss the impact of dispersion PPPM on the observed results. It was shown in Chapter 2 that using the PPPM method for dispersion provides simulations that are both more accurate and more efficient. From the perspective of computational efficiency, the role of the PPPM algorithm for dispersion can be easily identified: because simulations were faster, the simulation setup could be designed such that larger or longer simulations, or more different systems could be studied, which is of course beneficial, but was possibly not a critical requirement for the observation of the smooth contact line transition. What was much more important is that the usage of PPPM increased the accuracy of the simulations. The developed model for the surfactant could only reproduce both the large surface tension of pure water and the low surface tension of the surfactant laden interface by incorporating long-range interactions. Given that the superspreading effect is very sensitive to the choice of substrate and only occurs for a very narrow range of conditions it is likely that the corresponding mechanism could only be observed because our simulations provided a highly realistic representation and therefore were able to fall into this narrow range. If long-range dispersion interactions had not been incoroprated, this range might have been missed, the contact line mechanism would not have been observed, and the entire project would have been a failure.

3.4 Summary and concluding remarks

That previous MD studies on superspreading failed because of inaccurate molecular models motivated the development of an improved force field that accurately captures features assumed important for simulations of surfactants in general and the superspreading mechanism in particular. This new force field allowed to study the superspreading effect with large-scale MD simulations of surfactant enhanced spreading of aqueous solutions of different trisiloxane and alkyl ethoxylate surfactants on PP, PEO, and PTFE substrates. We cover simulation setups in which superspreading occurs in experiment, as well as simulations at various conditions that are unfavorable for superspreading, namely that the surfactant is too short or too long, the substrate is too hydrophilic or too hydrophobic, or because the surfactant does not belong to the general class of superspreading agents.

In simulations at superspreading conditions we observed a smooth transition at the contact line, which removes the sharp edge typically encountered in spreading droplets and thus overcomes the Huh–Scriven paradox. This offers an explanation for the greater spreading exponent observed in the viscous regime at superspreading conditions. An increased spreading exponent was also observed for aqueous solutions of trisiloxane droplets on a PEO substrate, which is too hydrophilic for superspreading. The increased spreading exponent for this simulation might be a result of water adsorbing at the substrate close to the contact line.

While the simulation scale is apparently too small to capture Marangoni stresses and therefore cannot directly observe the superspreading phenomenon, our simulation results provide insight about mechanisms at the contact line. The absence of bilayers, precursors, or feet casts doubt on the majority of the proposed superspreading hypotheses. In contrast, at superspreading conditions the droplet is bent inwards at the contact line and the transition from the liquid–vapor to the solid–liquid interface is smooth. This smooth transition renders possible direct surfactant adsorption through the contact line and in this way provides an explanation for the superspreading mechanism [107]. Based on this mechanism we provide plausible explanations for the complex dependency of superspreading on the substrate hydrophobicity, the surfactant chain length, and the surfactant concentration.

The simulation results reported here show a molecular mechanism at the contact line that potentially enables the fast wetting kinetics of superspreading and illustrates why the effect is not observed under conditions far from the optimum. However, these results do not explain the difference between conditions that both lead to superspreading, such as between two different substrates on which superspreading can occur or differnt trisiloxane surfactants with $n = 5$ and $n = 6$. Moreover, because of the larger length and time scales involved in Marangoni flows, the superspreading regime is not captured directly in our simulations. This also means that, although the proposed mechanism is simple, based on observations, and explains a lot of experimental results, there is no direct evidence that the observed mechanism is the molecular mechanism of superspreading. Further modeling to bridge the involved scales will be required to obtain a full understanding of all details of superspreading. Coarse-graining can provide further information on the role of vesicles and phase behavior. To fully capture the effect, however, accurate hybrid continuum and molecular simulation approaches that can cover both the molecular and the continuum scales will be required [98, 198].

4 Molecular precursors

4.1 Previous studies and open questions

Precursor films of molecular thickness that precede droplets were first discovered by Heslot et al. [19, 20]. Their study has motivated a series of experimental, theoretical, and numerical studies in which further properties of these films have been described. In the following we only provide a summary of the most relevant literature to the study at hand. In particular, we review the most elementary findings known from experiments, present the four theoretical models on molecular precursors whose current limitations show that the effect is not yet well understood and deserves further research, and describe previous MD studies. Afterwards, the objectives of this work are presented. A more complete overview of the available literature is given in [16, 199].

4.1.1 Experimental observations

In the first articles on molecular precursors, Heslot et al. [19,20] report the spreading of squalane, PDMS and tetrakis(2-ethyl-hexoxy)-silane droplets on (111) silicon wafers. Three different types of precursors, each with thickness on the length scale of molecular dimensions were observed. For the squalane droplet a continuously growing precursor was observed. The PDMS droplets were preceded by a layer with a constant width of approximately 6 Å, the thickness of a PDMS molecule, that smoothly becomes thinner at the end of the tip. For the tetrakis(2-ethyl-hexoxy)-silane droplet the striking phenomena of the evolution of several distinct layers, each with molecular-scale thickness, that spread ahead of the droplet, was discovered. This last phenomenon is also known as terraced wetting. As a fourth type of precursor we mention here a single layer precursor of molecular thickness with a sharp edge at the leading tip of the precursor described by Bardon et al. [200] for squalane droplets on the same surface at lower temperatures. The different observed precursor shapes are sketched in Figure 4.1. Many other types of observed precursors are summarized in [201].

In the overwhelming majority of experimental studies it was found that the dynamics

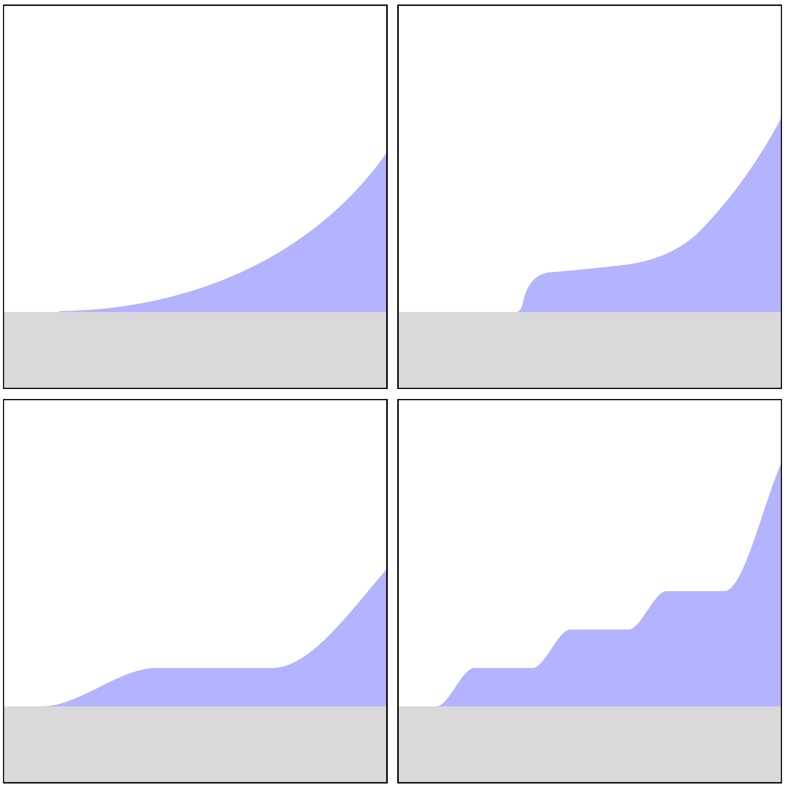


Figure 4.1: Sketches of different precursor types that have been observed experimentally. Upper left: Continuously growing precursor as observed for squalane [20]. Upper right: Sharp increasing and afterwards continuously growing precursor as observed for squalane at reduced temperature [200]. Lower left: A single layer precedes the droplet as observed for PDMS in [19]. Lower right: The terraced wetting effect in which multiple distinct layers precede the droplet as observed for tetrakis(2-ethyl-hexoxy)-silane.

of the precursor length l_p follows

$$l_p \approx (D_p t)^{0.5}, \tag{4.1}$$

i.e., has a diffusive behavior with a spreading exponent of 0.5, where D_p is a diffusion coefficient and t is the time. This scaling behavior was already observed in the first publication on molecular precursors [20] and was later confirmed in numerous studies (see [16] and references therein) with only one exception [202] that will be addressed later.

4.1.2 Theoretical models

Different models have been proposed to describe the evolution of the formation of layers. De Gennes and Cazabat [203] presented a hydrodynamic approach in which the precursor dynamics are described by competitive fluxes in horizontal and vertical directions. For multilayered spreading the model predicts a spreading exponent of 0.5 for each layer. For a single layer, however, this model predicts slower wetting dynamics. Moreover, the model requires distinct molecular layers of constant density and therefore cannot be used to describe continuously growing precursors. It should also be emphasized that whether one or more layers form is controlled by a friction coefficient; the formation of multiple layers is thus a result of the dynamic and not the thermodynamic properties of the system.

An alternative model by Abraham et al. [204, 205] is based on nonequilibrium statistical mechanics. The model expresses the free energy of a system as the sum of the free energies of distinct layers. Compared to the model discussed in the previous paragraph, the existence of a single or multiple precursors does not depend on the friction in the system, but depends only on thermodynamic quantities. The friction coefficient in this model only influences the dynamics with which the layers develop. This model predicts the occurrence of precursors if the spreading coefficient is positive, so that the system is in the complete wetting regime. Similar to the model by de Gennes and Cazabat, a weakness of this model is that the spreading exponents observed in experiment can only be reproduced in certain limits and that a growing precursor cannot be described because of the assumption of distinct layers of constant thickness.

The first microscopic model of precursor formation was developed by Burlatsky et al. [206, 207]. This model describes the development of a monolayer of fluid on a substrate that is connected to a bulk reservoir on one end and has a free end on the opposite site. The particle movement is described via molecular jumps between adjacent lattice positions. This model predicts the correct spreading exponent of 0.5 for the monolayer. The development of the monolayer occurs if a microscopically defined spreading coefficient s is positive, which is equal to minus the energy of moving a particle from the bulk reservoir to the tip of the film. The model, and extensions in [208], are sufficiently simple to allow for approximate solutions that can be used to describe the system. Shortcomings of this model are that it describes only a monomolecular layer, and not terraced or growing precursors, and that the microscopic spreading coefficient s might be useful when developing a theoretical model, but its determination or prediction in reality can be difficult.

Finally, an Ising model was presented in [209] and [210]. In addition to the model of Burlatsky et al, this model allows the existence of a second molecular layer and mass transport between the two layers. This model also provides the correct spreading exponent for the first layer and is sufficiently simple to allow for approximate analytic solutions. Moreover, it can be extended for the description of multilayered spreading, as discussed in [16]. A major drawback of the current version of this model is that it does not predict under which conditions precursors form.

As briefly described above, none of the existing models is capable of capturing all of the relevant experimental findings. The first models based on a continuum description of layers of constant density did not successfully capture the film dynamics, suggesting that the continuum representation of the layers is oversimplified. The models based on a microscopic description provide the correct exponents, but are not yet in a state in which the different kinds of precursors can be captured appropriately. Further studies are thus required to better understand the phenomenon and develop improved models.

4.1.3 Findings from molecular dynamics simulations

That MD simulations are an ideal method to study molecular precursors because of the length scales inherent to the phenomenon has been realized early and the first MD study on the subject by Yang et al. [211] appeared only two years after the discovery of molecular precursors. This and a study from the same authors [212], however, could not reproduce Equation (4.1). The discrepancy was later resolved by Bekink et al. [213], who showed that the scaling behavior observed in [211, 212] was caused by a porous solid in their simulations. Because of computational limitations, the studied systems were restricted to small system sizes, containing only a few thousand atoms each of liquid and substrate. In all three studies the authors report of having performed simulations in the terraced regime, whereas from the results they presented there is no clear evidence for the formation of multiple distinct layers either in the depicted snapshots or the plots of the width of the different layers.

The studies by Nieminen et al. [214, 215] that appeared a few years after the first articles by Yang et al. [211, 212] were performed with even smaller systems. In their works they vary the shape of the solid–liquid pair potential, and examine lower temperatures, such that the simulated liquid is nonvolatile. The spreading exponent known from experiment was reproduced in their studies. Moreover, from the images of the provided simulation snapshots it seems that terraced layers formed, as two distinct layers separate from the droplets, possibly because of the lower temperatures. Unfortunately the systems were run with very small number of particles and the authors did not measure the layer width except the width of the first one. Another weak point that might have severe effects on the reported results is that the authors use a flat substrate as a model and simply use a Nosé-Hoover thermostat for the fluid. The roughness of the solid and any friction effects, which are typically considered of critical importance [216], are neglected, which possibly had a strong influence on the results.

Finally, the temperature $T = 0.57/k_B$ used in parts of their simulations is far below the triple point $T \approx 0.7/k_B$ of the atomistic LJ fluid [217]. Their observations could thus be a result of that the system was not in a liquid state.

The first simulation studies in which the authors support their claims of having observed the terraced wetting effect by providing analysis also for higher layers than only the first layer above the substrate are those reported by Banavar and co-workers [218, 219]. These authors report spreading simulations of chain molecules on crystalline solid substrates. The droplets in their study are already considerably larger, but still only contain either 16 384 or 32 000 particles. Because of the three-dimensional system setup, the majority of the simulated particles, 147 456, are part of the solid substrate. Aside from simulating larger systems, an additional improvement over previous studies is that the spreading exponent of the precursors is determined from a fit to the data in [218]. It was found that the spreading coefficient was not exactly 0.5, but was approximately 0.46. The evidence for having captured the terraced effect is rather weak. In [218] there are only pictures of the snapshots from which a distinct separation of the layers is not observable. In [219], similar snapshots are shown in which the formation of distinct layers is not visible; additionally they provide the number of atoms in the first four layers above the substrate. It is argued that the number of particles in the layers increases linearly with time to prove the existence of multiple layers; the depicted lines, however, are clearly not straight lines, which makes their claims questionable.

That the terraced wetting effect was captured in the studies of Banavar and co-workers was also put in question by studies presented by Heine, Grest, and Webb almost a decade later [186, 191, 220], where simulations similar to those mentioned in the previous paragraph were performed with much larger systems. The formation of multiple distinct layers was not observed and it was concluded that capturing the effect in MD simulations was infeasible because of computational limitations. The progress made in these studies occurs on many fronts. First, the simulated droplets were sufficiently large that properties of the main part of the droplet, i.e. the part of the droplet that is not the precursor, and the precursor can be studied at the same time. Second, they compare findings for the spreading velocity of the main part of the droplet to spreading models. Third, which will be addressed in more detail in the following, the authors explored how spreading simulations can be performed at reduced computational cost.

In the MD studies discussed here prior to those of Heine et al., the authors either used a flat substrate and neglected the effect of friction, or they modeled the substrate with particles, resulting in the substrate particles outnumbering the number of fluid particles (e.g., by a factor of more than 4 in the studies by Banavar and co-workers [218, 219]).

Simulating the behavior of the substrate is thus the most demanding task in these simulations, which is a problem since the features of interest is the behavior of the liquid. This issue is addressed in [191], where simulations are performed with particle-based crystalline substrates and perfectly flat substrates. In the simulations of perfectly flat substrates the friction is modeled by a clever choice of thermostats. It is shown that when using the thermostats properly, the spreading velocities on flat substrates is equal to what is observed on crystalline substrates. It should be noted, though, that the crystalline substrate induces ordering in the liquid layers close to the substrate [212] that cannot be captured with their approach. To further increase computational efficiency, Heine et al. [186] studied the effect of using cylindrical instead of spherical droplets. They observed that, with a proper translation of the findings between the 2D and 3D geometries, the physics of the simulated systems are identical, and that simulations can therefore be performed with a cylindrical setup to save computation time.

4.1.4 Objectives

As outlined above, there was a considerable effort of studying precursors with MD simulations. Features that have been treated especially well in even the earliest studies [211, 212, 218, 219] included how the simulated systems should be prepared and how the dynamics should be controlled by thermostats. In contrast, little effort has been devoted to characterizing the shape of the precursors. Proof for the claims of what kind of precursor shape (cf. Figure 4.1) was captured in the simulations was usually only provided with figures of simulation snapshots [211,214,218] or linear fits to clearly non-linear measured data [219], which is rather weak evidence and possibly also the reason for the discrepancies between earlier [218, 219] and more recent studies [186, 191, 220]. Another flaw of previous simulation studies is that the precursor length in all MD studies that we are aware of (all referenced in [16] or [199]), incorrectly measured the precursor as the distance r from the precursor tip to the center of the droplet (either by measuring this distance or indirectly by counting the number of atoms in each layer), whereas the correct measure should be the distance l from the tip of the precursor to the macroscopic contact line of the droplet, as shown in Figure 4.2. The development of an improved analysis method that facilitates a clear identification of the simulated precursor type and the identification of which findings from previous simulation studies should be questioned because of the incorrect precursor measurements are the first objectives of this study.

The only experimental study in which significant discrepancies from the diffusive

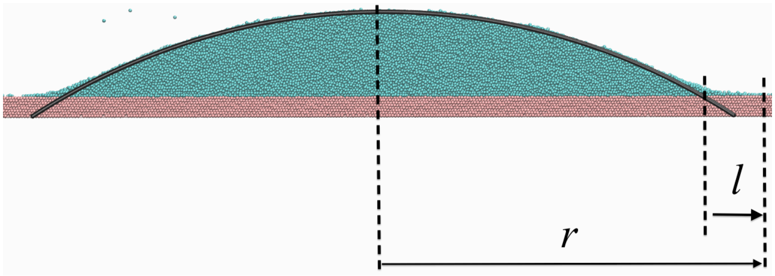


Figure 4.2: Liquid droplet (cyan) on a solid substrate (pink). Correct definition of the precursor length l and incorrect measure r that has often been employed.

layer dynamics in Equation (4.1) has been reported, to the best of our knowledge, is the study by Albrecht et al. [202], in which the spreading kinetics of extremely small PDMS droplets with a volume of only a few picoliters was studied. Significantly smaller spreading exponents of only 0.12 and 0.14 were reported. The authors argued that the decreased spreading exponent was a result of the hindrance of diffusion caused by entanglement of chain molecules. The PDMS molecules that were used in their study, however, had an average molecular weight $M_w = 760\,\mathrm{g/mol}$, which corresponds to an average chain length of 10 or 11 monomers, for which entanglement effects should not be that pronounced, as shown in [221], where the normal diffusive regime with a spreading exponent of 0.5 was recovered in bulk simulations of linear polymer chains with 100 monomers and more. Moreover, this finding also conflicts with Heslot et al. [20], who observed diffusive behavior in experiments with longer chains of PDMS. As previous MD studies, Albrecht et al. [202] took a measure that relates the precursor length to the distance from the film tip to the droplet center. The second objective of this study is to show that the discrepancy between the study of Albrecht et al. [202] and other experimental studies is possibly a result of the size of the droplets used in [202] in combination with the incorrect measure of the precursor length.

While the dynamics and mass transport mechanisms of molecular precursors seem to be well understood, conditions under which precursors form and what causes the development of different types of observed precursors (cf. the sketches in Figure 4.1) are two of the most important remaining open questions in the field of precursors [16]. The available models for precursors are unable to address these issues, and findings from experiment and simulation studies provide no definite answers either. For the information under which conditions a precursor film can develop, there are conflicting findings. It has been found in numerous experimental and simulation studies that the onset of a precursor matches the transition from the partial to the complete wetting regime when the spreading coefficient becomes positive [186, 191, 216, 222–224]. This conflicts with the lack of theoretical justification for this relation and with other studies that reported precursors in the partial-wetting regime [16]. Information that describes under which conditions different precursor shapes are observed is absent. The third objective is to resolve the conflicting findings and explain under which conditions precursors can form and what controls the different observed precursor shapes.

To address these questions we performed spreading simulations of droplets of atomistic fluids and chain molecules. The simulation setup, analysis, and results of these simulations is described in Section 4.2. The impact on the results of the inaccurate measures for the precursor is also discussed. The requirements for precursor formation are addressed in Section 4.3 with free energy computations. Final remarks are given in Section 4.4.

Before presenting the results, we briefly comment on the incorrect measurement of the precursor length in previous studies. The reasons why these incorrect measures were chosen is that most experimental studies, in which the droplets are large and the exact definition of the precursor films therefore does not seem to play a dominant role, use r as the length of the precursor. Precise information that the length of the precursor is only the distance from the precursor tip to the macroscopic droplet edge is given rarely in the literature and only appeared in recent years [16, 225, 226].

4.2 Simulations of molecular precursors

4.2.1 Simulation setup

We perform MD simulations of spreading droplets. Based on the findings of Heine et al. [186], who found that simulations of cylindrical droplets show the same behavior as spherical droplets, we used cylindrical droplets in our simulations. Instead of capturing the effect of the substrate by an effective potential and a thermostat as done in their simulations, we explicitly model the beads of the crystalline substrate. Our motivation for treating the solid explicitly is that ordering in the liquid structure induced by the solid crystal might effect the details of the properties of the precursor layers. Indeed, when comparing Figures 1 and 2 from [191], in which the simulations on explicit and implicit substrates are compared, the simulation results look qualitatively similar, a closer inspection however suggests that the precursor layer formed on the explicit substrate is thinner. As pointed out above, modeling the substrate explicitly has the disadvantage of increasing the computation time. For cylindrical droplet geometries, as those used here, however, the ratio of solid to liquid particles is much lower than for spherical simulation setups, as shown below. The computational overhead thus becomes more tolerable.

Spreading simulations were performed with cylindrical droplets with the axis of the cylinder pointing in the y-direction. The substrate's surface normal pointed in the z-direction, so spreading occurs in x-direction. Simulations were performed with periodic boundary conditions in all directions. While periodicity in the y-direction is required because of the cylindrical droplet geometry, periodicity in the x and z-direction was required only because of PPPM for dispersion. Because our focus is not on describing the features of a specific molecule but understanding the underlying physics, we perform simulations with model substances and use reduced units in all simulations in this chapter. All data are reported with respect to a reference energy ϵ, mass m, distance σ, and derived time unit $\tau = (m\sigma^2/\epsilon)^{1/2}$.

The substrate in all spreading simulations was a fcc crystal with a unit length of 1.5874σ and (111) vector normal to the surface pointing in the z-direction. The substrate was composed of 8 layers, the lowest layer of which was held rigid in all simulations. For the spreading simulations, the substrate extended 600.747σ and 33.6739σ in the x- and y-directions and contained 142 772 particles. The liquid droplet was composed of 180 000 beads. The number of substrate particles in the simulations is still large. Because of the cylindrical setup, however, their number is at least smaller than the fluid particle number instead of being significantly larger. The initial configuration of the droplet was created using PACKMOL [71].

In simulations of chain molecules, interactions between two beads i and j are described with the LJ potential

$$U_{\mathrm{LJ}} = 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right], \tag{4.2}$$

where r_{ij} is the distance between the beads, ϵ_{ij} is the depth of the potential well and σ_{ij} is the distance at which the potential passes through zero. For the substrate-substrate interactions we use $\epsilon_{ss} = 5\epsilon$ and $\sigma_{ss} = \sigma$, which provides a stable solid at the simulated temperature. The liquid–liquid interactions are described with $\epsilon_{ll} = \epsilon$ and $\sigma_{ll} = \sigma$. For the solid–liquid interactions, we use $\sigma_{sl} = \sigma$. Different values of ϵ_{sl} are used in the simulations. In addition to the LJ potential, bonded atoms interact with the FENE potential [227]

$$U_{\mathrm{FENE}} = -0.5kr_0^2 \ln\left[1 - \left(\frac{r_{ij}}{r_0}\right)^2\right], \tag{4.3}$$

where r_0 is the maximum possible length of the bond and k is a force constant. We use $k = 30\epsilon/\sigma^2$ and $r_0 = 1.5\sigma$. Each linear chain molecule consisted of 20 beads. The solid particles have reduced mass $m_s = 2m$ and the liquid particles $m_l = m$.

The main motivation to use chain molecules in this and previous studies is to perform simulations with a fluid of low volatility to exclude effects caused by droplet evaporation and condensation of material from the vapor onto the substrate. In contrast, simulations with atomistic fluids would be desirable because their theoretical description is much easier. In fact, the two theoretical models capable of describing the dynamics of the first layer properly [206, 210] are both designed for atomistic fluids. Moreover, atomistic fluids can be handled more easy in molecular simulation than chain molecules both in terms of performing simulations and extracting data. The problem with simulations of atomistic fluids is, however, that for normal LJ fluids, the vapor phase has a non-negligible density, so the fluid is volatile above the melting temperature [228]. The challenge of avoiding volatility for atomistic systems has previously been addressed by running simulations either at temperatures below the melting point [214], or with two immiscible liquids, one of which corresponds to the spreading droplet and one which models the environment and ensures nonvolatility [222]. We propose here an alternative procedure to model an approximately nonvolatile atomistic fluid. In simulations with the atomistic fluid, the interactions between particles i and j is described by a modified Buckingham potential [229]

$$U = \frac{\epsilon_{ij}}{1 - \frac{6}{\lambda_{ij}}} \left\{ \frac{6}{\lambda_{ij}} e^{\left[\lambda\left(1 - \frac{r_{ij}}{\sigma_{ij}}\right)\right]} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^6 \right\} + \frac{D_{ij}}{r_{ij}^{12}}, \tag{4.4}$$

where r_{ij} is the distance between i and j, and λ_{ij} controls the shape of the curve. If $D_{ij} = 0$, the original Buckingham potential, where ϵ_{ij} and σ_{ij} are the depth and the position of the potential well, is recovered. The last term avoids the unphysical maximum for small r_{ij} in the original Buckingham potential. For the substrate we use $\epsilon_{ss} = 5\epsilon$, $\lambda_{ss} = 12$, $\sigma_{ss} = 1.122\sigma$, and $D = 0.05\epsilon\sigma^{12}$. For the liquid we use $\epsilon_{ll} = \epsilon$, $\lambda_{ll} = 9$, $\sigma_{ll} = 1.1\sigma$, and $D_{ss} = 0.03\epsilon\sigma^{12}$. For the solid–liquid interaction we use $\lambda_{sl} = 12$, $\sigma_{sl} = 1.1\sigma$, and $D_{sl} = 0.03\epsilon\sigma^{12}$. ϵ_{sl} is varied in the simulations. As for the chain simulations, we use $m_s = 2m$ and $m_l = m$ for the mass of the solid and liquid particles.

The substrate particle interaction was chosen such that a stable solid phase is formed at the simulated temperature. The liquid–liquid parameters were selected such that interactions are softer at shorter particle separations compared to an LJ potential, as shown in Figure 4.3. As a result, the melting point of the liquid is reduced and simulations can be performed at lower temperatures, which results in less fluid volatility. Effects due to evaporation and condensation can therefore be neglected. That the fluid has low volatility is apparent from a snapshot of a slab simulation of the atomistic liquid shown in Figure 4.4. The condensed phase is liquid; the vapor phase is almost empty. For simulations with the pure liquid in the slab setup, crystallization did no occur within simulations over $20\,000\tau$. Crystallization of the droplet material occurred in few simulations when brought in contact with the solid substrate. When crystallization has occurred, this is given explicitly when presenting the simulation results and the influence of solidification on the reported results is discussed.

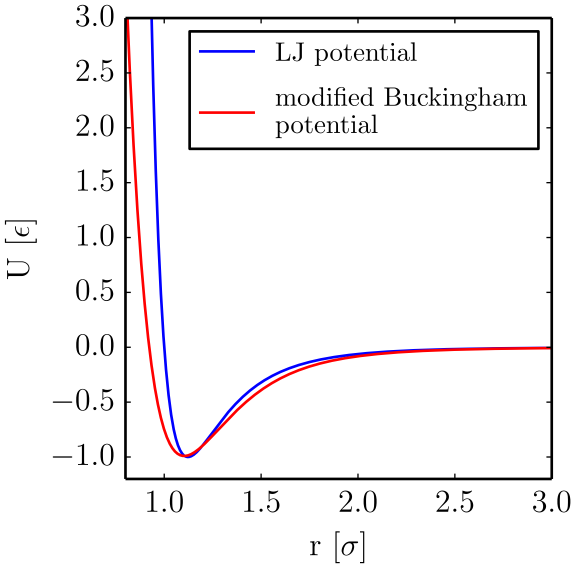


Figure 4.3: LJ and modified Buckingham potential for the atomistic fluid. The Buckingham potential is less repulsive at lower particle distances.

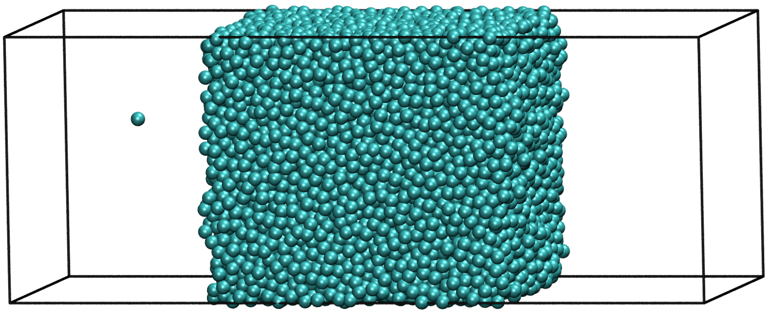


Figure 4.4: Equilibrated slab system of the atomistic fluid. The condensed phase is liquid and the vapor phase is almost empty.

For simulations with the chain molecules, the substrate was equilibrated at $T = \epsilon/k_B$ using a Langevin thermostat [167] with damping factor $\tau_D = 0.3\tau$. The droplet was preequilibrated for 500τ at a reduced temperature of $T = \epsilon/k_B$ using a Nosé-Hoover thermostat [68] with damping factor $\tau_T = 0.5\tau$. For the atomistic fluid, the substrate and droplet were preequilibrated for 50τ and 1000τ at a reduced temperature of $T = 0.6\epsilon/k_B$ using a Nosé-Hoover thermostat [68] with damping factor $\tau_T = 0.5\tau$.

For both, the atomistic fluid and the chain molecules, the equilibrated droplet was then positioned 2.5σ above the equilibrated substrate. The first five substrate layers above the rigid layer were coupled to a Langevin thermostat [167] with a damping factor $\tau_D = 0.3\tau$. No thermostat was used for the rest of the simulation, as in [218], because it provides closest similarity to experiment.

During the equilibration and production, the particle positions and velocities were updated using a two-stage rRESPA integrator [83], with interactions within a distance of 3.0σ shifted to zero starting at 2.5σ and computed every 0.005τ. Long-ranged dispersion interactions were computed using a PPPM solver with $i\mathbf{k}$ differentiation, interpolation order $P = 5$, grid spacing $h \approx 0.75\sigma$, Ewald parameter $\beta = 1.0/\sigma$ and a real-space cutoff $r_c = 3.0\sigma$ and were evaluated every 0.02τ. The repulsive terms were truncated at r_c. The spreading progress is visualized in Figure 4.5.

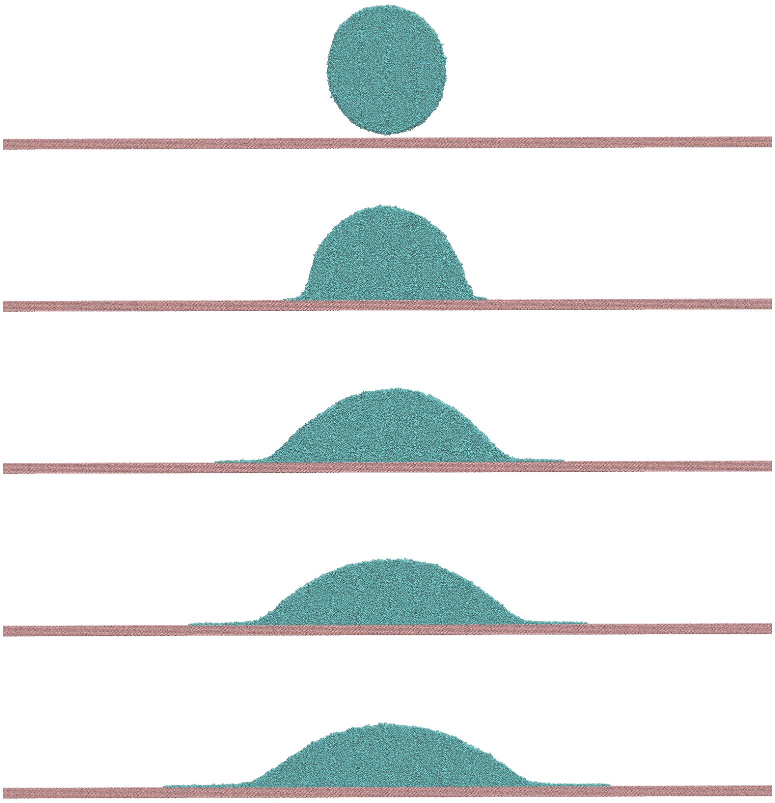


Figure 4.5: Spreading process of chain molecules (cyan) on a substrate (pink) with $\epsilon_{sl} = 1.5\epsilon$ at times (from top to bottom) 0τ, 1250τ, 5000τ, 7500τ, and 10000τ. A thin precursor evolves.

4.2.2 Analysis methods

4.2.2.1 Layer classification

Obviously, if the target is to define the requirements for precursor formation and different types of precursors, the first necessary step is to extract from the simulation if and what kind of precursor has formed. To address this target we first identify molecular layering of the liquid in the vicinity of the substrate. The length of each layer ahead of the droplet is then determined in a second step. The development of these widths over time in combination with additional analysis reported below can be used to characterize the simulated precursor.

The identification of layers is done with the same approach used in the first MD studies on precursors [212]. Near a solid wall, such as the substrate in our simulations, liquid particles are more ordered compared to the bulk liquid and form layers of molecular thickness. The separation of these layers can easily be determined from density profiles of the liquid in z-direction, as shown in Figure 4.6. For small values of z, the profile oscillates, with each peak corresponding to a molecular layer. The position of the minima are used as separators between different layers. With increasing distance from the substrate the oscillations decay. Defining discrete layers is not possible for this larger distance (but is also not required because relevant processes in the context of molecular precursors occur close to the substrate).

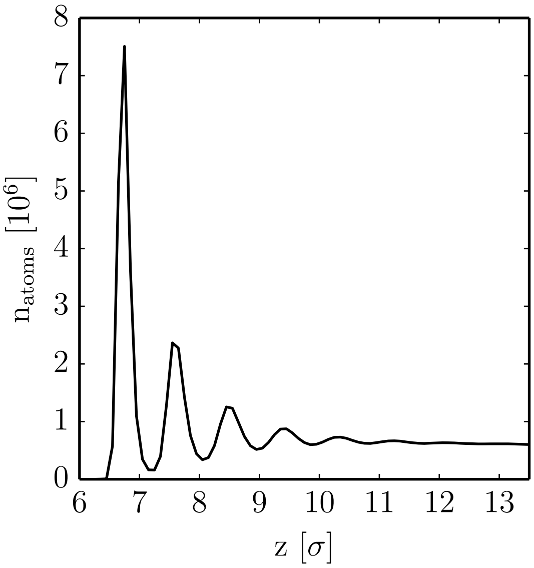


Figure 4.6: Number count of liquid particles over the z dimension. For low separations from the substrate, there is a natural layering of the atoms, each corresponding to one molecular layer.

The next step is to define the width r of each molecular layer. Previous approaches were to approximate the width of a layer by the distance between the minimum and maximum extrema of the particles [214], the number of particles included in that layer [218, 219], or the point where the local binned density drops below a threshold [191, 211]. The first approach obviously fails if single molecules separate from the droplet and spread far ahead of the contact line and the rest of the precursor. The second approach is more stable, but changes in the local density of the precursor, such as those reported in [212], cannot be captured by this method. The last approach compensates for this weakness, but the position of the interface can only be resolved to the width of the bins used to create the histograms of the densities. To overcome these shortcomings, we use an alternative approach to determine the width of each layer. For a given snapshot, the particles are binned in a histogram in the x and z directions. For small z, the bins are separated according to the separation of layers, for large z, where layering is not visible, the bins are separated by a distance σ. The width of the bins in the x direction is uniformly taken as σ. Each horizontal layer of the histogram was fit to an error function. The inflection point of the error function is used as the position of the liquid–vapor interface, which for the lowest layers corresponds to the tip of the precursor.

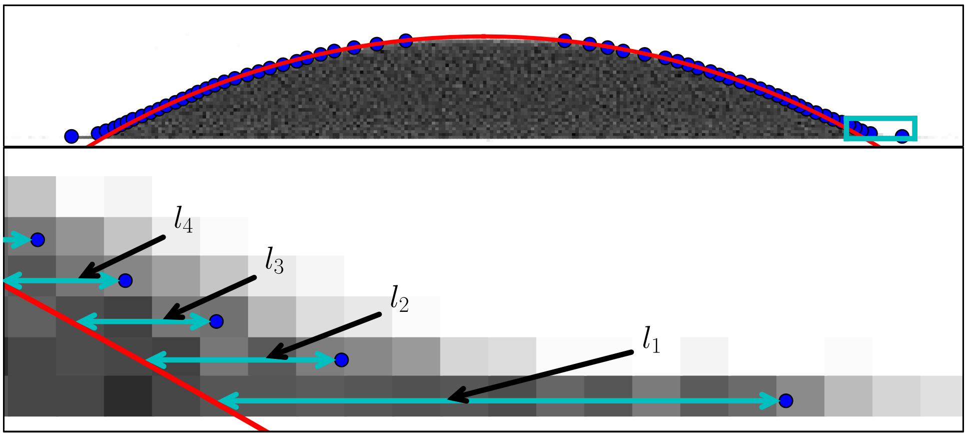


Figure 4.7: Definition of the precursor length. Instantaneous droplet positions are binned in a histogram. The interface position (blue points) is determined from fits of error functions to the histogram. A circular arc is fitted through the upper 20 pairs of the interface position. The precursor length in each layer is the horizontal distance from the blue points to the red curve. Top: Entire droplet; bottom: region in the cyan rectangle from the upper image with the first layers' length l_i.

The final step in the analysis is to determine the width of the precursor for each layer. As mentioned above, previous studies have incorrectly measured the precursor length as the distance from the precursor tip to the center of the droplet, whereas the width from the tip of the precursor to the macroscopic contact line of the droplet is the correct measure [16]. To define this distance, we fit a circular arc to the 20 upper pairs of the positions of the interface, where 20 was chosen to provide good fits to the macroscopic droplet shape while avoiding inclusion of the region close to the foot. The horizontal distance of the liquid–vapor interface to the circular arc is the precursor length l_i of layer i. The horizontal distance to the liquid–vapor interface to the center of the drop (the incorrect value for the layer width used in previous studies), is denoted as r_i in the following. A sample histogram with the measured interface, a circular fit, and the definition of l_i is shown in Figure 4.7.

The new method presented thus far has the benefit of clearly defining whether layers separate from the droplet and how many of them separate. If multiple layers separate from the droplet, however, the method cannot be used to identify the shape of the precursor, as shown in Figure 4.8. Continuously growing and terraced precursors can provide identical evolution for different layers l_i over time. Additional analysis is thus required to fully characterize the precursor.

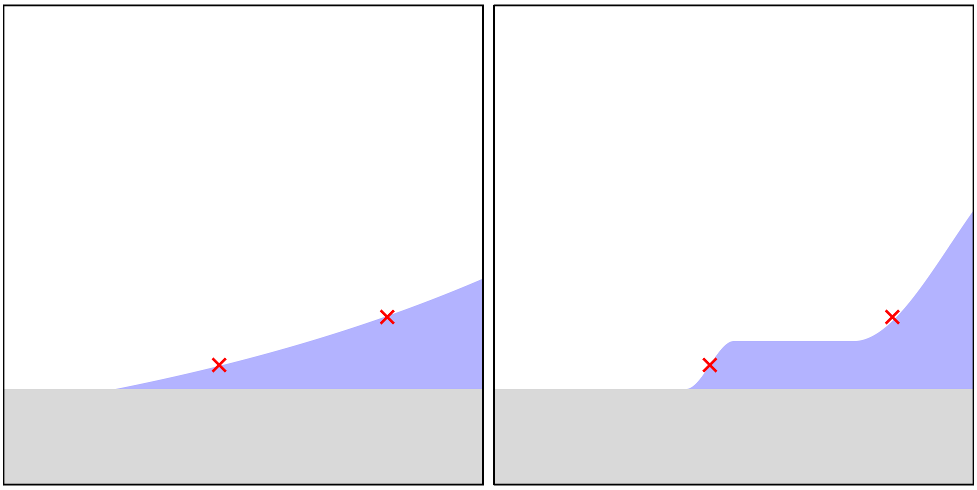


Figure 4.8: Measuring the precursor length l_i does not provide a complete description of the precursor, but only provides the position of the red crosses. Distinguishing, for example, between a continuously growing precursor as on the left and a precursor with a well defined layer as on the right is not possible.

One approach would be to use snapshots of the simulations to distinguish between these states. However, this approach does not provide strong evidence and has led to conflicting statements on the type of precursor observed in previous studies. The information can also be extracted from the histograms of the density (cf. Figure 4.7), but this still leaves room for discussion. We propose to extract the missing information by determining the shape of the droplet similar to what is obtained in ellipsometric experiments with which the effect of molecular precursors was discovered first. Ellipsometric measurements provide an effective thickness of the liquid that can, for a dilute layer, be smaller than the molecular diameter [16] and corresponds to the average density of the fluid. This quantity can be approximated in simulation by binning the particles in the x-direction only and counting the number of particles in each bin. The obtained data is then scaled such that the maximum value of the histogram corresponds to the height of the droplet at its center. The resulting data is similar to the results obtained from ellipsometric measurements and can be used as an auxiliary tool to differentiate precursor types. Figure 4.9 shows an example of this analysis.

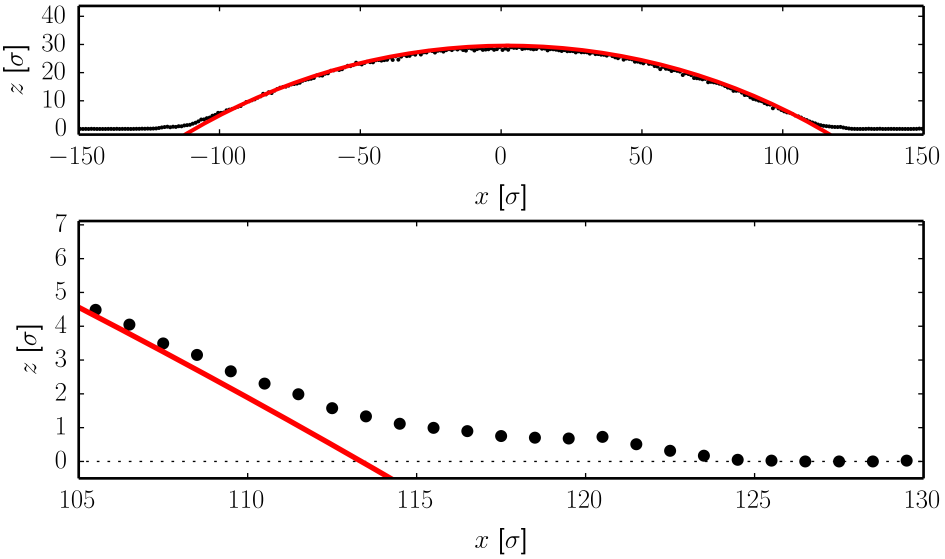


Figure 4.9: Measurement of the droplet profile to mimic ellipsometric experiments. Black circles: measured droplet height, red line: circular fit to the droplet profile determined from the histograms of the density. Top: Entire droplet with equal scaling in x and z dimensions. Bottom: selected region that highlights the precursor with unequal scaling in x and z dimensions to capture the different dimensions. The substrate surface is at $z = 0$. Measurements of the droplet profile with $z = 0$ as on the right of the lower image correspond to a density of zero.

4.2.2.2 Determination of spreading exponents

Spreading exponents can be determined for the correct and incorrect measures of the precursor length l_i and r_i. For the data that is obtained for r_i, a spreading exponent can be determined by fitting a function of the type

$$f_r = C(t - t_0)^{\alpha}, \tag{4.5}$$

to the measured data, where C is a coefficient, t_0 is a reference time that needs to be subtracted from the results because the position of $t = 0$ is not defined in the simulation, and α is the spreading exponent. The parameters C, t_0, and α are determined in the fit. The fit was performed using orthogonal distance regression, which is superior to simple least square minimization for the type of function above because of the strong gradient of the function for values of t close to t_0. The region close to t_0 dominates the fit, whereas the region at larger times is more important to compute an accurate value of the spreading coefficient, especially because for small t the simulation results might be influenced by noise caused by that the droplet not yet entering the proper spreading regime.

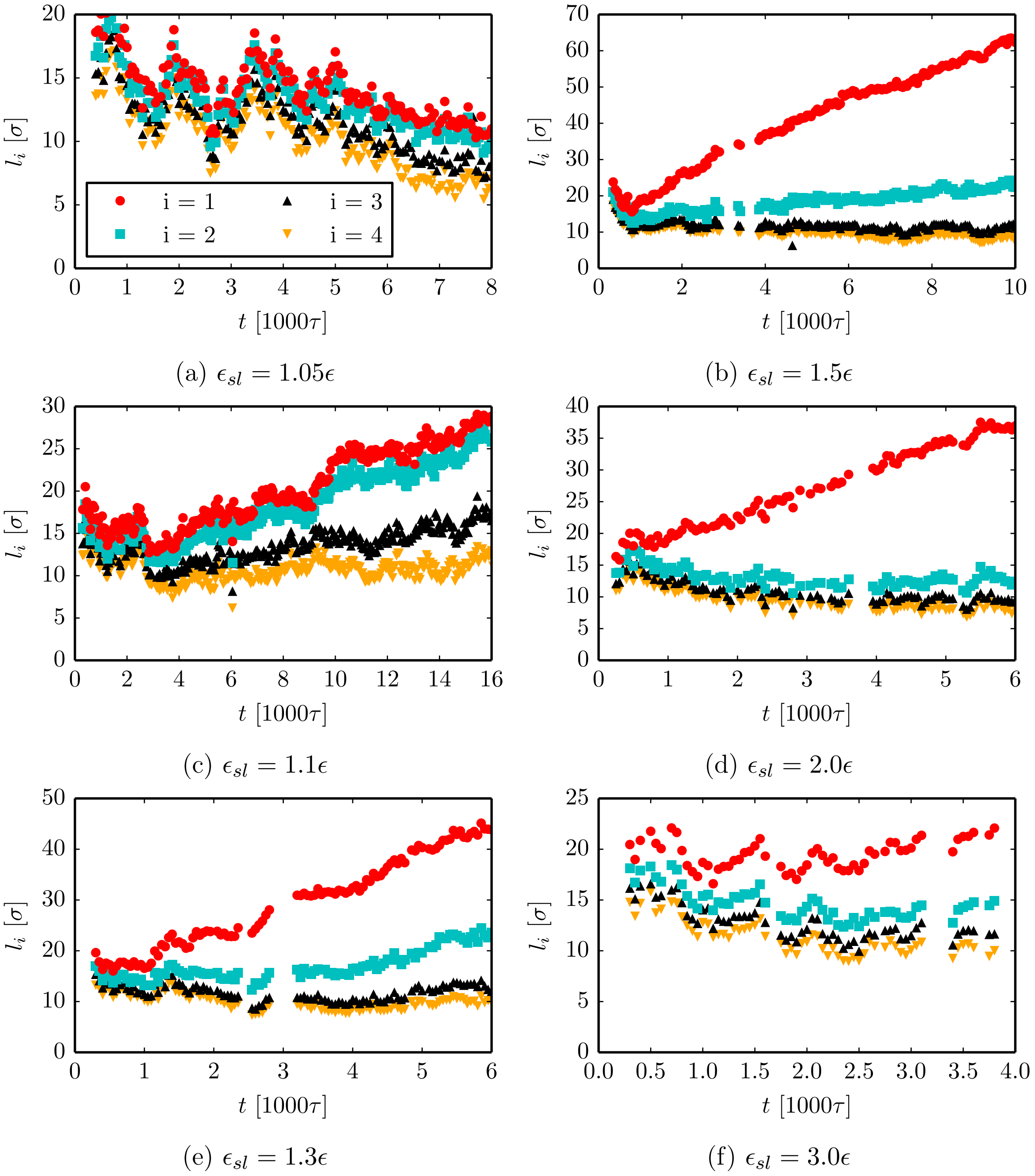
For the correct value of the precursor width l_i, the spreading exponents are much more difficult to determine. l_i is computed using the difference between the tip of the drop and the width of the circular arc at the z position of the layer i. Because of this, the values for l_i are subject to strong noise. Moreover, the growth of l_i is much slower than that of r_i. As a consequence, determining the spreading exponent α for this data as above for the incorrect measure r_i does not work. Instead, we make a linear fit to the log–log representation of the data. Further difficulties and possibilities to overcome them are presented below.

4.2.3 Precursor types, transition regimes, and wetting dynamics

4.2.3.1 Precursor types and transitions for chain molecules

Results for the time evolution of the length l_i of the first layers of the polymer droplets for different values of ϵ_{sl} are given in Figure 4.10. It is immediately apparent from the results that for $\epsilon_{sl} = 1.05\epsilon$ no precursor is formed, whereas for all other depicted images a precursor that separates from the droplet is formed. For all examined values of $\epsilon_{sl} < 1.05\epsilon$ no precursor was formed either. The energy $\epsilon_{sl,c}$ where the onset of the precursor occurs is $1.05\epsilon < \epsilon_{sl,c} < 1.1\epsilon$. Values for ϵ_{sl} between these values were not examined.

For $1.1\epsilon \leq \epsilon_{sl} < 2.0\epsilon$, it is clear that not only the first layer above the substrates separates from the droplet. As discussed above, however, these figures do not provide evidence to which precursor type the layers should be attributed. From the effective densities of the droplet shape given in Figure 4.11 it becomes apparent that distinct layers cannot be detected in any simulation. It is thus evident that for the simulations of the chain molecules there is a continuously growing and not a terraced precursor. This conclusion is in agreement with that by Heine et al. [186, 191, 220] and questions results in [218, 219] where terraced wetting was reported for a similar setup.



(a) $\epsilon_{sl} = 1.05\epsilon$

(b) $\epsilon_{sl} = 1.5\epsilon$

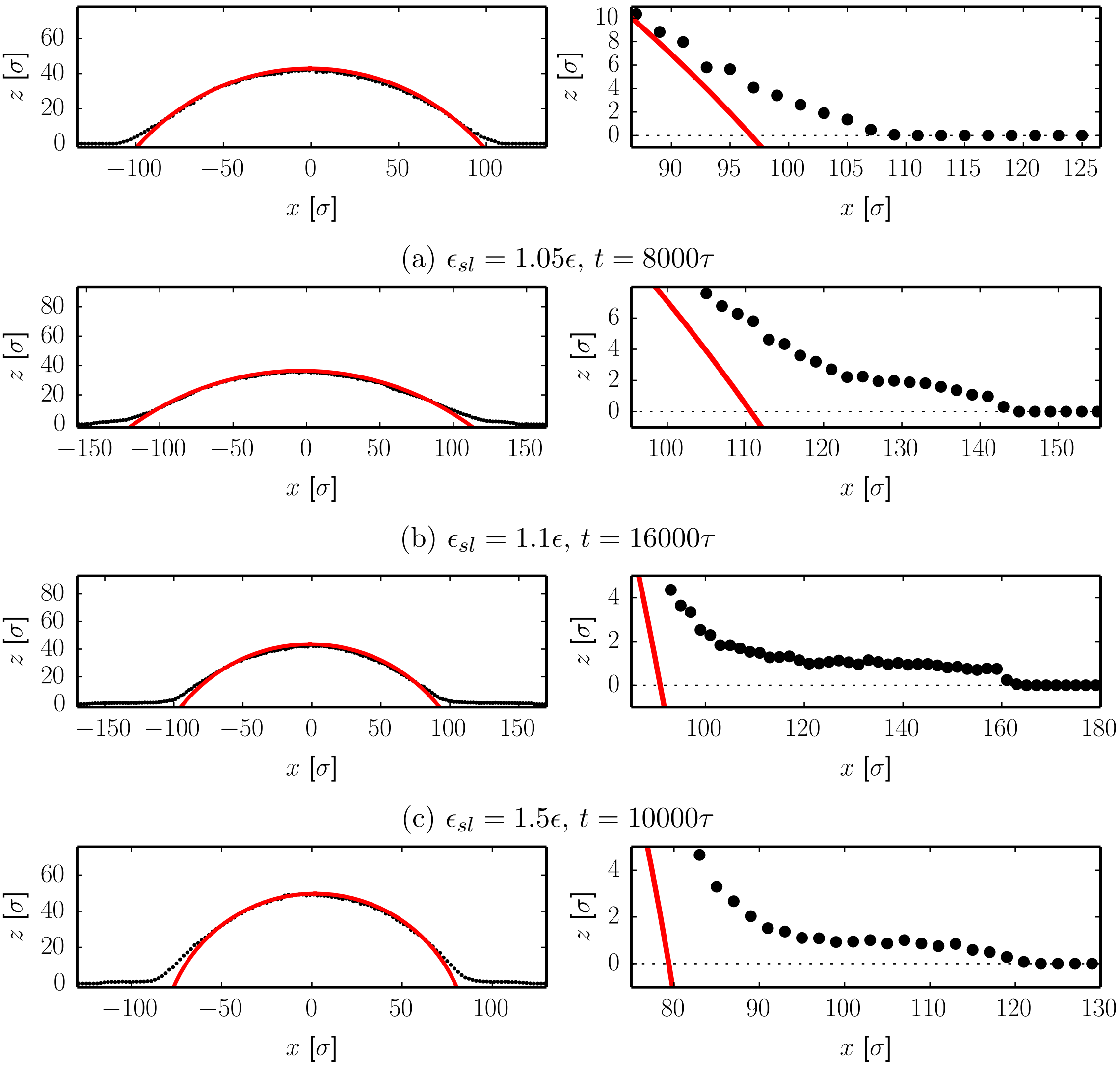
(c) $\epsilon_{sl} = 1.1\epsilon$

(d) $\epsilon_{sl} = 2.0\epsilon$

(e) $\epsilon_{sl} = 1.3\epsilon$

(f) $\epsilon_{sl} = 3.0\epsilon$

Figure 4.10: Evolution of the first four layers of the chain molecules at different solid–liquid interactions ϵ_{sl}. For $\epsilon_{sl} = 1.05\epsilon$, no precursor forms, whereas layers separate from the droplet for stronger solid–liquid interactions. For intermediate substrate energies, multiple l_i grow over time, whereas only a single layer grows for the strongest interactions.



(a) $\epsilon_{sl} = 1.05\epsilon$, $t = 8000\tau$

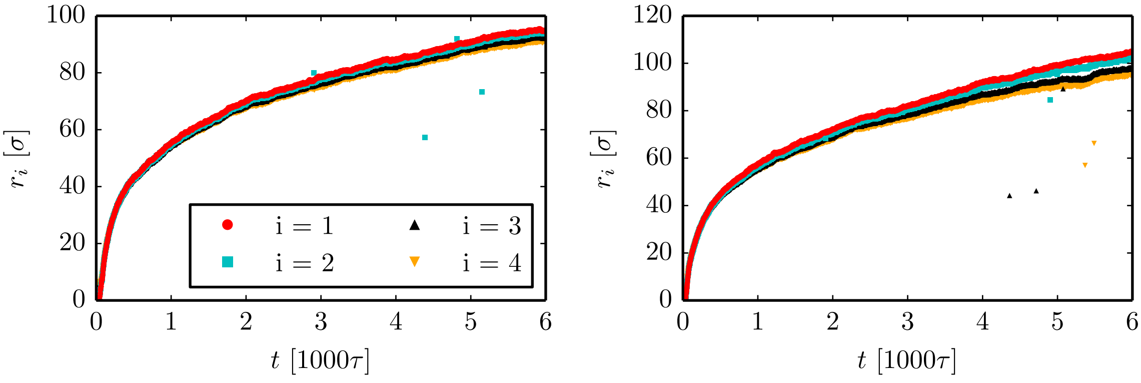
(b) $\epsilon_{sl} = 1.1\epsilon$, $t = 16000\tau$

(c) $\epsilon_{sl} = 1.5\epsilon$, $t = 10000\tau$

(d) $\epsilon_{sl} = 2.0\epsilon$, $t = 6000\tau$

Figure 4.11: Effective thickness of the droplet of chain molecules for different values of ϵ_{sl}. Left: entire droplet with equal scaling in x and z directions. Right: selected region with different scaling. The snapshots correspond to the final snapshot from each simulation. Multiple distinct layers do not occur in simulations with chain molecules. While it seems from Figure 4.10 that multiple distinct layers separate, results in these images show that the precursor is continuously growing and not terraced.

The properties and dynamics of the precursor change with increasing ϵ_{sl}. As can be seen from Figure 4.10, the spreading velocity of the first layer increases up to $\epsilon_{sl} = 1.5\epsilon$ and then starts to decrease again. For the highest simulated value of $\epsilon_{sl} = 3.0\epsilon$ the rate of layer growth is very slow. These observations agree with results in [216], where the maximum spreading velocity of the precursor at intermediate substrate energies is explained by a balance of driving forces and friction forces between the layer and the substrate. The width of the second layer l_2 shows a similar behavior as the first one, with the difference that the maximum spreading velocity is observed for $\epsilon_{sl} = 1.3\epsilon$. The separation of the layer from the main droplet is essentially no longer visible for substrate energies $\epsilon_{sl} \geq 2.0\epsilon$. The separation of a third layer from the droplet is only visible for $\epsilon_{sl} = 1.1\epsilon$ and is not visible for any higher values of ϵ_{sl}. Overall, these findings show that as long as the friction between the lowest layer and the substrate does not dominate spreading, the precursor layer becomes more slender with increasing ϵ_{sl}, which can be explained by a competition of particle displacements in vertical and horizontal directions. The stronger the interaction between the substrate and the liquid, the greater the tendency for the fluid to move to the first layer, which thins the precursor. These observations support the idea of de Gennes and Cazabat [203], whose model is based on competitive horizontal and vertical displacements.



(a) $\epsilon_{sl} = 1.0\epsilon$

(b) $\epsilon_{sl} = 1.1\epsilon$

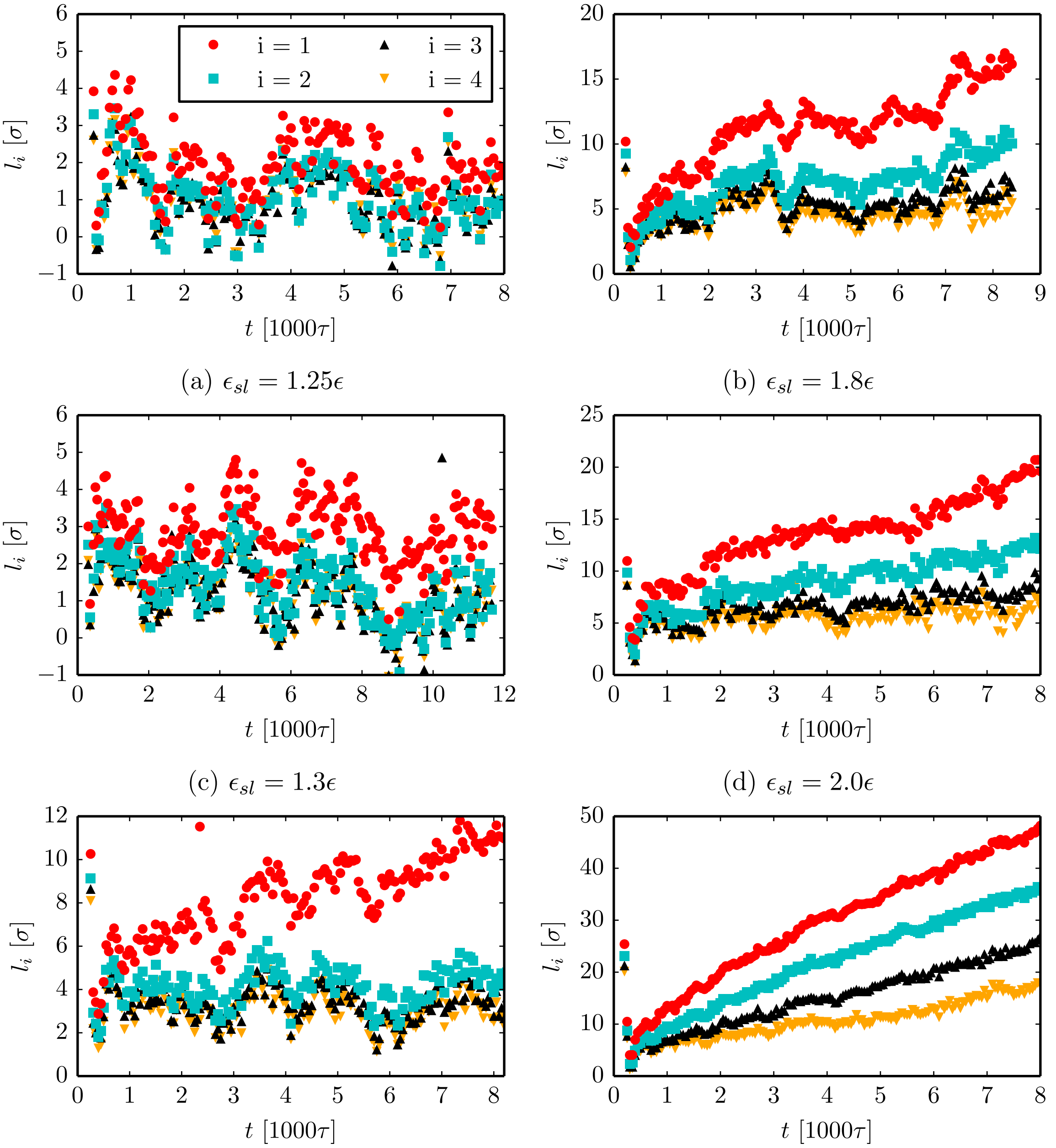
Figure 4.12: Evolution of the value or r_i of the first four layers of the chain molecules. l_i in Figure 4.10 shows that no precursor forms for $\epsilon_{sl} = 1.0\epsilon$ whereas a precursor forms at $\epsilon_{sl} = 1.1\epsilon$. This finding is hardly visible from this image.

Before proceeding with the atomistic fluid, we briefly comment on the capability of distinguishing precursor regimes by means of the incorrect measure for the layer width r_i. Figure 4.12 shows the evolution of r_i of the first layers of chain molecules for $\epsilon_{sl} = 1.0\epsilon$ (no precursor regime) and $\epsilon_{sl} = 1.1\epsilon$ (continuously growing precursor regime). While from Figure 4.10 discrimination between the different spreading regimes is possible, it is not visible from the evolution of r_i over time that a precursor develops for the higher substrate energy.

4.2.3.2 Precursor types and transition for the atomistic fluid

The evolution of the length of the first layers l_i over time for the atomistic fluid is shown in Figure 4.13 for different ϵ_{sl}. Similar to the chain molecules, no precursor forms for the lowest substrate energies. In contrast to the chain molecules, however, the atomistic fluid passes through different precursor regimes. For $\epsilon_{sl,c1} < \epsilon_{sl} < \epsilon_{sl,c2}$, where the exact value of $\epsilon_{sl,c1}$ and $\epsilon_{sl,c2}$ will be discussed later, a single layer separates from the droplet. That only a single layer separates can be seen best for $\epsilon_{sl} = 1.6\epsilon$ in Figure 4.13 where l_i only grows for the first layer. This finding is also confirmed by results of the effective density in Figure 4.14 (although the visibility of the effect is here less apparent than from Figure 4.13), which shows that a single layer slowly grows, stays constant for a few σ, and then has a transition to the main part of the droplet where the thickness starts to increase again. When the substrate energy increases further, such that $\epsilon_{sl} > \epsilon_{sl,c2}$, a second transition occurs, in which multiple layers separate from the droplet. This regime can be best identified for $\epsilon_{sl} = 2.0\epsilon$ from Figures 4.13 and 4.14. Figure 4.13 shows that for $\epsilon_{sl} = 2.0\epsilon$, multiple layers separate from the droplet, and Figure 4.14 reveals that there is a continuously growing layer rather than multiple distinct layers. For the highest value of $\epsilon_{sl} = 3.0\epsilon$ used in this study, a fourth regime is sampled. As for $\epsilon_{sl} = 2.0\epsilon$, multiple layers separate from the droplet in Figure 4.13. In contrast to the results for $\epsilon_{sl} = 2.0\epsilon$, however, the precursor is not continuously growing but has distinct steps, as can be seen from Figure 4.14. This suggests that the terraced effect was captured in this simulations. Visualization of the simulation, however, revealed that the droplet partly crystallized. The effect was thus possibly observed because of the transition to the crystalline phase of the droplet material. This is in agreement with the study of Nieminen et al. [214], where the simulation was run at temperatures below the melting point. Because of the solidification of the droplet material for this value of ϵ_{sl}, this finding was not studied further. We note that crystallization of the fluid was observed in this spreading simulation exclusively.

The transition between the regime of a single layered precursor and the continuously growing precursor can be seen directly from the time evolution of the different layers in Figure 4.13. For $\epsilon_{sl} = 1.6\epsilon$ only a single layer separates from the droplet. For $\epsilon_{sl} = 1.8\epsilon$, l_i grows for multiple layers over time. No further values of ϵ_{sl} were sampled between those values. The transition between these two regimes thus occurs at $1.6\epsilon < \epsilon_{sl,c2} < 1.8\epsilon$. The onset of a single layer precursor is harder to identify from the time evolution



(a) $\epsilon_{sl} = 1.25\epsilon$

(b) $\epsilon_{sl} = 1.8\epsilon$

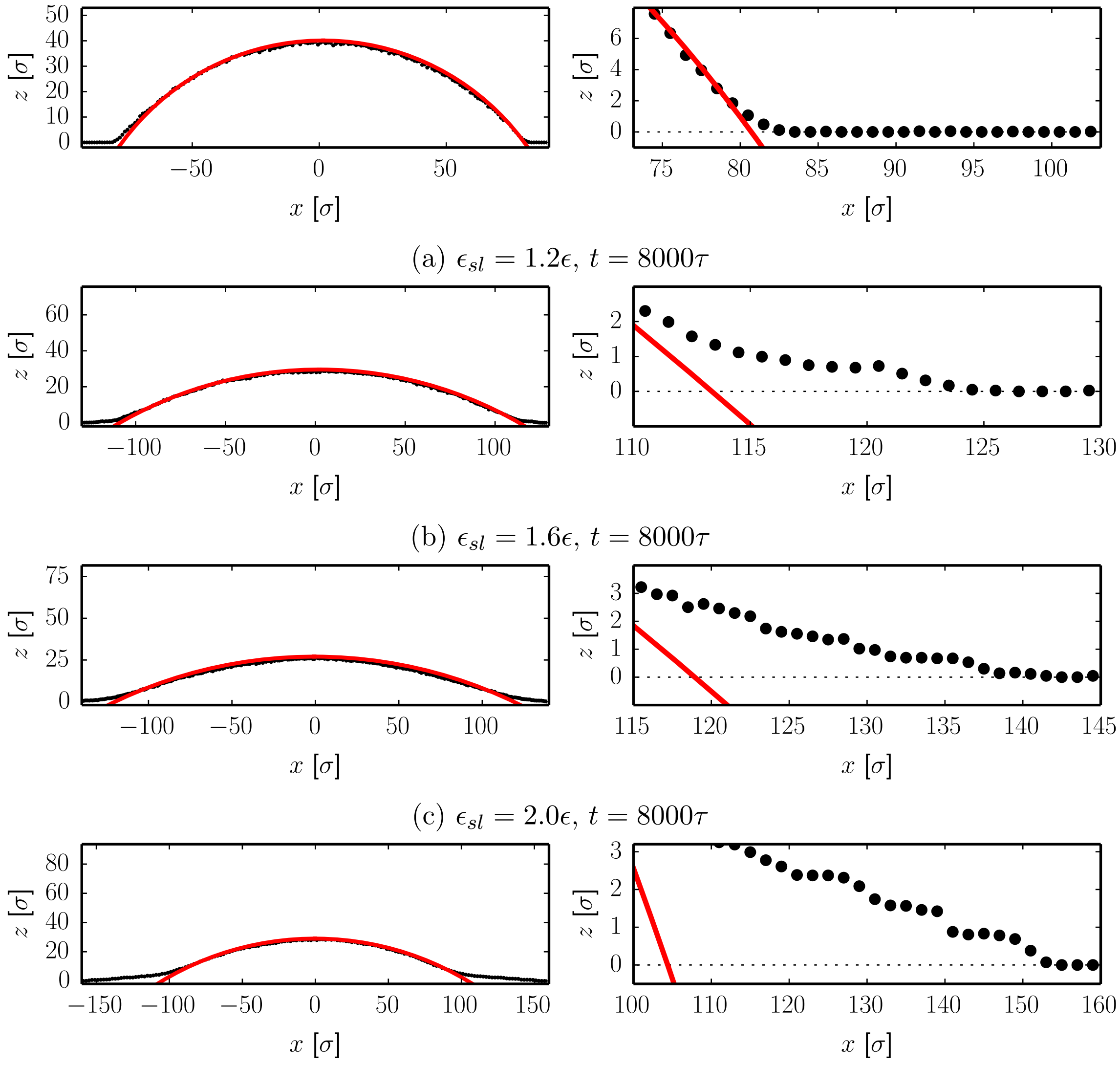
(c) $\epsilon_{sl} = 1.3\epsilon$

(d) $\epsilon_{sl} = 2.0\epsilon$

(e) $\epsilon_{sl} = 1.6\epsilon$

(f) $\epsilon_{sl} = 3.0\epsilon$

Figure 4.13: Evolution of the first four layers of the atomistic fluid at different solid–liquid interactions ϵ_{sl}. For low substrate energies, no precursor forms. For intermediate energies at $\epsilon_{sl} = 1.6\epsilon$ a single layer separates. Multiple layers separate for substrate energies $\epsilon_{sl} \geq 1.8\epsilon$.



(a) $\epsilon_{sl} = 1.2\epsilon$, $t = 8000\tau$

(b) $\epsilon_{sl} = 1.6\epsilon$, $t = 8000\tau$

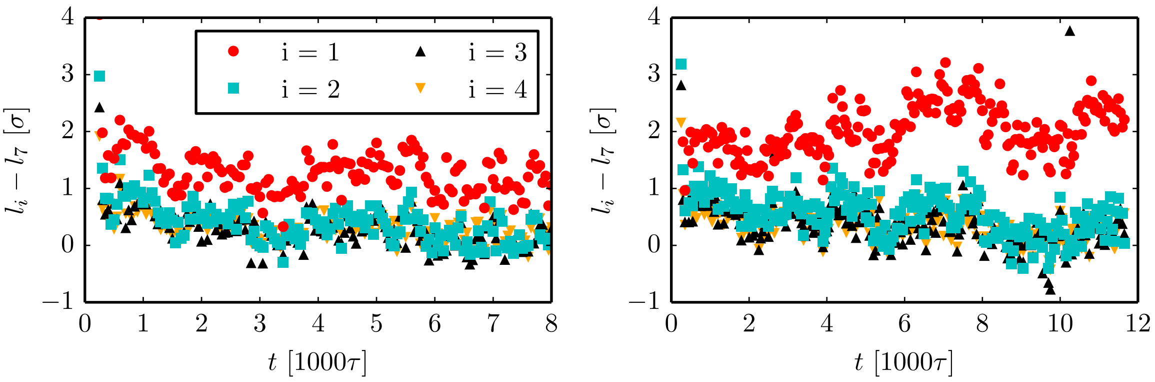
(c) $\epsilon_{sl} = 2.0\epsilon$, $t = 8000\tau$

(d) $\epsilon_{sl} = 3.0\epsilon$, $t = 8000\tau$

Figure 4.14: Effective thickness of droplets of atomistic fluid for different values of ϵ_{sl}. Left: entire droplet with equal scaling in x and z direction. Right: selected region with different scaling. The snapshots correspond to the final snapshot from each simulation. No precursor forms at the lowest substrate energy. For $\epsilon_{sl} = 1.6\epsilon$, a single layer forms. (That this is not a continuously growing precursor is not obvious from this image but becomes clear in combination with Figure 4.13, where only a single layer separates from the droplet). A continuously growing precursor forms for $\epsilon_{sl} = 2.0\epsilon$ and precursor with multiple distinct steps at $\epsilon_{sl} = 3.0\epsilon$.

of the layer width. Close to the transition from no precursor to a single layer precursor, the first layer grows very slowly and is thus harder to identify. An additional difficulty is that the data is very noisy. The sources of the noise are the intrinsic thermal noise of MD simulations and the uncertainty of fitting the droplet layers to error functions. The dominant reason for the strong fluctuations, however, is that the layer width must be determined from a difference between the outer tip of the layer and a circular fit to the droplet, which is again subject to uncertainties. As a result, the transition cannot be directly identified from the time evolution of the layer width l_i in Figure 4.13.

What is well visible in Figure 4.13 is that the noise in the data follows a trend. When the the first layer passes through a peak, all other layers simultaneously pass through a peak. These trends are a result of the noise introduced by the circular fit. When the fitted circular arc grows over time, all layers shrink; when the circular arc shrinks, all layers grow. Because this deviation occurs for all layers, the noise in the data can be reduced by subtracting the width of a higher reference layer. Results are shown in Figure 4.15 for $\epsilon_{sl} = 1.25\epsilon$ and $\epsilon_{sl} = 1.3\epsilon$. While for $\epsilon_{sl} = 1.25\epsilon$ no layer separates from the droplet, the slow separation of the first layer from the droplet can be observed for $\epsilon_{sl} = 1.3\epsilon$. The transition between the regime of no precursor and the single layer precursor thus occurs as $1.25\epsilon < \epsilon_{sl,c1} < 1.3\epsilon$. No further values of ϵ_{sl} were sampled in that interval.



(a) $\epsilon_{sl} = 1.25\epsilon$

(b) $\epsilon_{sl} = 1.3\epsilon$

Figure 4.15: Evolution of the first four layers minus a reference layer. Because the precursor evolves slowly close to the crossover point, the transition between no precursor and a single layer precursor cannot be identified directly from Figure 4.13. Subtracting a reference layer reduces the noise of the circular fit, such that it becomes visible that l_1 slowly separates for $\epsilon_{sl} = 1.3\epsilon$ whereas no layer separates from the drop for $\epsilon_{sl} = 1.25\epsilon$.

4.2.3.3 Measured spreading exponents

In this section we first address the spreading coefficients as determined from the “incorrect” measure for the precursor length r_i. The spreading exponents obtained from the time evolution of l_i will be discussed later. Figure 4.16 shows the time evolution of the tip-to-center distance of the first layer above the substrate r_1 for the simulations with the chain molecules and varying ϵ_{sl}. In addition to the measured data, depicted as points, the plot contains the fitted functions as continuous curves. Aside from the velocity of the precursor tip being fastest for intermediate substrate energies ($\epsilon_{sl} = 1.5\epsilon$), one can see that the resulting fit of Equation (4.5) works well for both the droplets in the regimes without a precursor ($\epsilon_{sl} = 1.0\epsilon$) and with a precursor (all other lines).

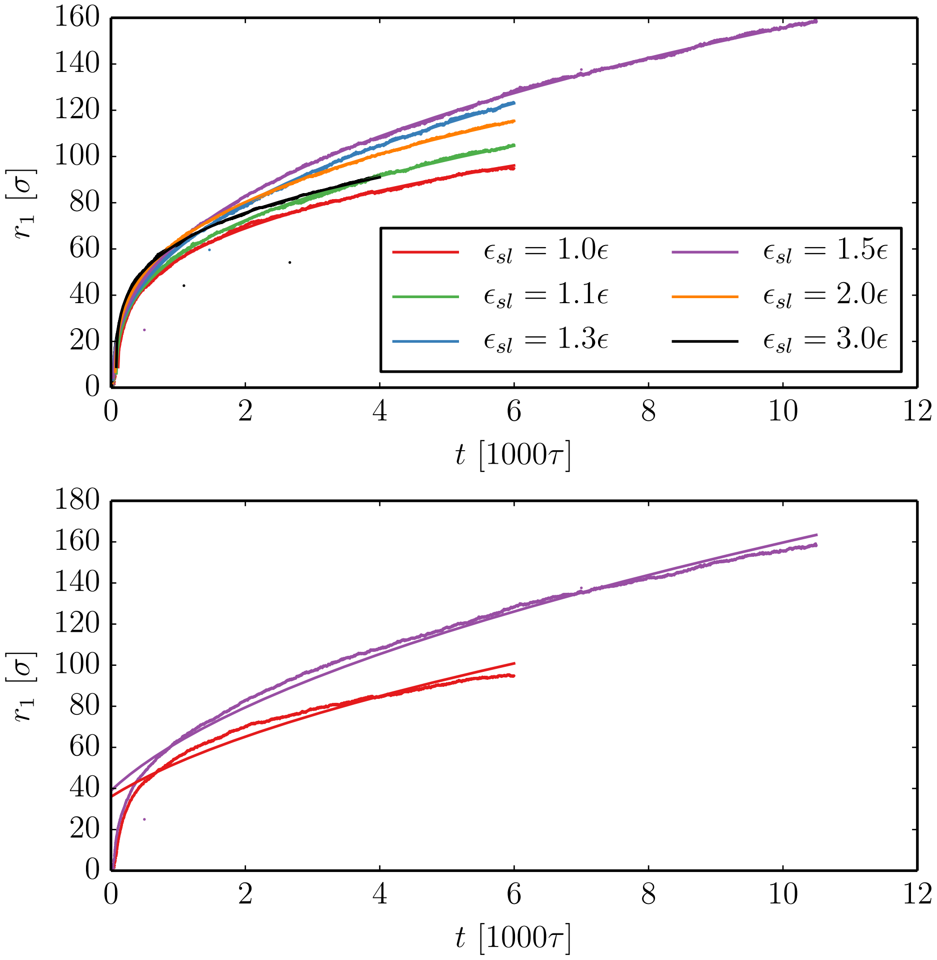


Figure 4.16: Measured values of r_1 and fitted functions for the chain molecules. Scatter points (that almost look like lines because of their high number): measured data. Continuous lines: fitted functions. Top: fit without constraints. Bottom: fit with constraint that $\alpha = 0.5$. When α is not constrained, excellent agreement can be achieved between the measurements and the fit. Enforcing $\alpha = 0.5$ provides a lower quality fit.

The spreading exponents that result from the fit are depicted in Figure 4.17 as a function of the solid–liquid interactions ϵ_{sl}. With increasing substrate energy, the measured exponents increase, reaching a maximum of $\alpha \approx 0.4$ for $\epsilon_{sl} = 1.3\epsilon$, and then start to decrease again. The measured spreading exponent is neither a constant nor is it close to the exponent of 0.5 expected for diffusive spreading. The good agreement between the fit and the measured data suggests that the specified spreading exponents are accurate. For reference, we have also tried to perform the fit by enforcing a spreading exponent of 0.5 in the lower image of Figure 4.16. The quality of the fit is much poorer. These results show that the evolution of r_i does not necessarily have a spreading exponent of 0.5. That diffusive growth was reported in previous simulation studies for the incorrect measure of the precursor length r might be a coincidence or that the spreading exponent was not measured carefully enough.

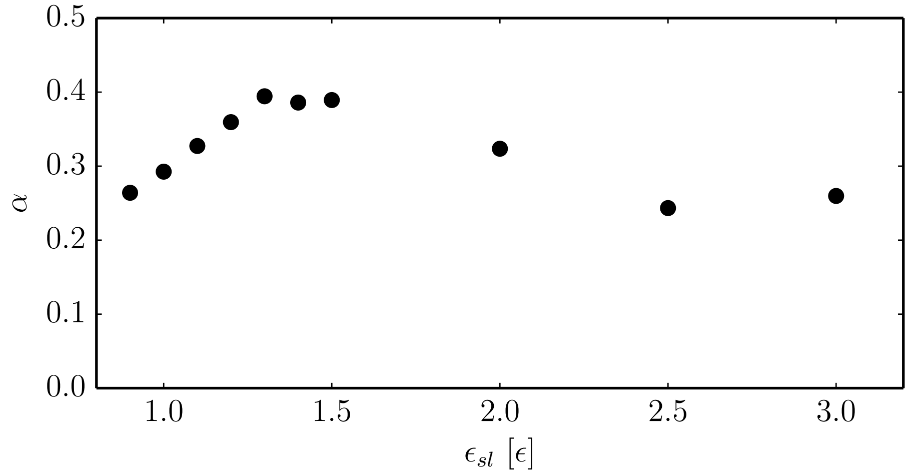


Figure 4.17: Spreading exponent obtained from the fit of measured values of r_1. α is distinctly smaller than the diffusive exponent of 0.5 for all values of ϵ_{sl} and is dependent on the substrate energy.

We now address the spreading exponents obtained when analyzing the evolution of the correct measure for the precursor length l_i. As discussed above, a determination of the spreading exponent from the data in linear representation as for r_i is impossible because of the slow evolution of the films and the noise in the data. The linear and the log–log data of the time evolution of the first layer l_i of the simulations with the chain molecule for three different values of ϵ_{sl} are given in the top and middle image of Figure 4.18. As can be seen from the log–log data, all three precursors seem to approach the correct diffusive behavior with increasing simulation time.

The gradient of the fitted line to the log–log data is the spreading exponent. Although performing a linear fit to a set of points is numerically trivial, determining the spreading

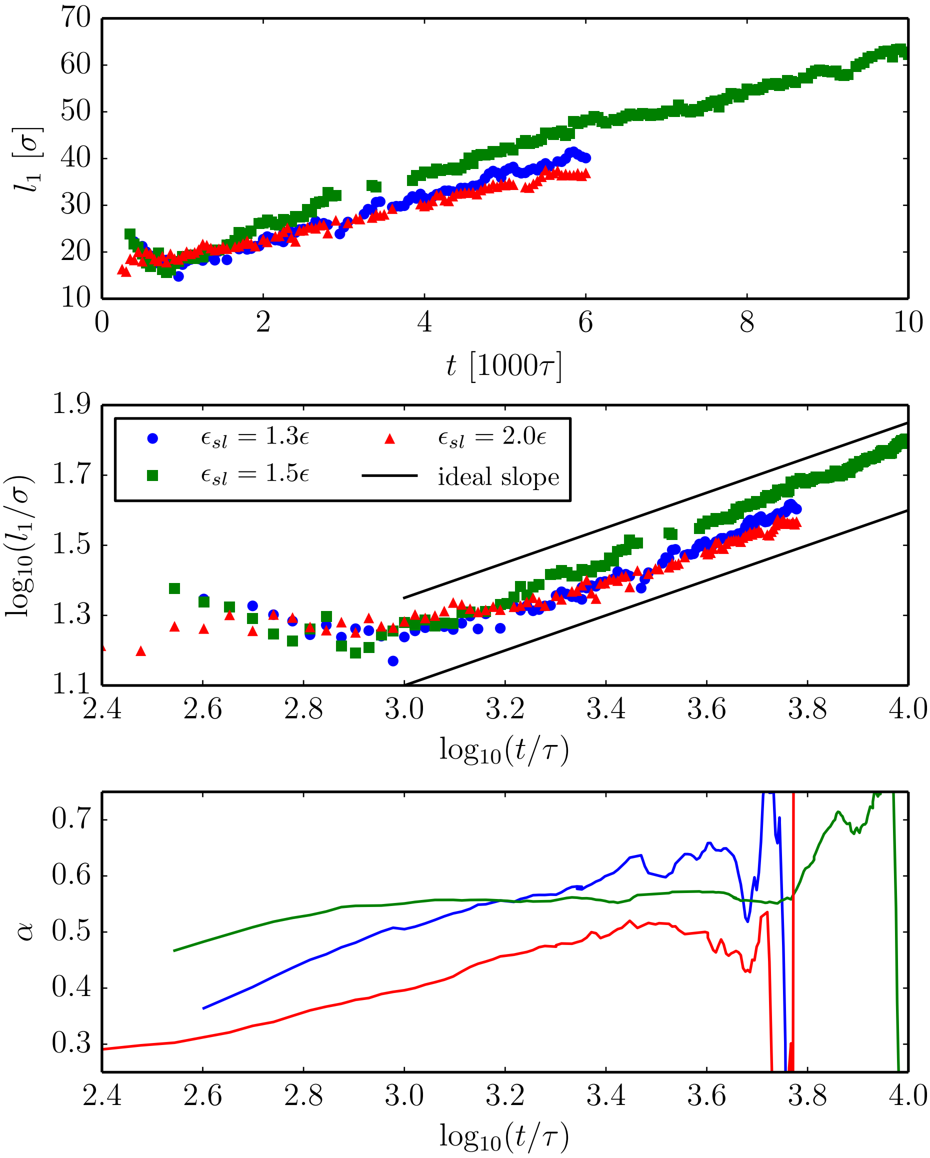


Figure 4.18: Analysis of the spreading exponent from the correct measure of the precursor length l_1. Top: measured data in linear representation. Middle: measured data in log–log representation with straight lines with the ideal slope of 0.5 that matches the diffusive spreading exponent. The measured data seem to approach the diffusive behavior for long simulation times. Bottom: Attempts to measure the spreading exponent from linear fits to the log–log representation. The desired plateau is only observed for $\epsilon_{sl} = 1.5\epsilon$ and is located at $\alpha \approx 0.55$.

exponent properly is still challenging, because the starting time t_0 is undefined and can impact the results. Moreover, data at the beginning of the simulation is unusable because the precursor has not yet developed and should be excluded. When excluding too much data, however, the data set for the fit becomes too small and the noise in the data dominates the fit. While we cannot offer a solution to overcome the undefined starting time t_0, we can suggest an approach to handle the amount of data to be included in the fit. To determine how much data should be selected, we performed the fit for different choices of the selected data: for each data point of l_i over t, we performed a fit such that only the data points at higher values of t are included, whereas all points at lower values of t are excluded. In this way we determine a different value for the spreading exponent at each value of t. For small values of t, the spreading exponent is influenced by the data from the beginning of the simulation being included. For large t, the spreading exponent is controlled by the uncertainty caused by the data set being too small. If the approach has worked successfully, the spreading coefficients form a plateau for intermediate values of t. The height of the plateau approximates the spreading exponent. The resulting development of the spreading exponent for different data sets is shown in the lower image of Figure 4.18. The fitted value for α forms a plateau for $\epsilon_{sl} = 1.5\epsilon$, whereas no plateau is formed for other values of ϵ_{sl}. The height of the plateau is approximately $\alpha = 0.55$, which is in reasonable agreement with the expected value of 0.5. The remaining difference could be a result of that the starting time t_0 is not well defined. We would like to point out that the plateau was observed only for this single spreading simulation. All other simulations were thus either too short or the data too noisy to properly quantify the spreading exponent and the correct spreading behavior can only be confirmed from the log–log representation.

4.2.4 Effects of correct and incorrect precursor analysis

We here suggest a novel approach to characterize precursors in simulations of droplet spreading. Instead of incorrectly measuring the distance from the droplet tip to the droplet center, as done in previous studies, we use the correct measure from the droplet tip to the “macroscopic” contact line. In addition, we compute effective densities similar to what is obtained in ellipsometry studies. When combining the results from the evolution of the different layers and the effective densities, a clear identification of whether a precursor is formed and a characterization of the shape of the developed precursor is possible. It should be noted that both approaches are required to fully describe a precursor. Only using the time evolution of the layers cannot distinguish between continuously growing precursors and terraced layers. Likewise, using the effective densities does not provide an accurate measure for the transitions between different precursor regimes and cannot identify whether a precursor separates or not. A disadvantage of the method seems to be that the determination of the spreading exponents with this approach is difficult, because a circular fit has to be performed to identify the macroscopic contact line and also because the layer width l_i grows much slower than r_i. The latter is not a problem of the described method, but a feature of the precursors, though.

The incorrect measure for the precursor width r_i has several severe disadvantages. First, unlike the method presented here, a determination of the precursor regime or even whether a precursor is formed is not possible. Second, the observed spreading dynamics of r_i do not correctly reflect the spreading dynamics of a precursor. While the majority of the conclusions drawn in previous studies with the incorrect measure for the precursor are in agreement with what we find despite the incorrect measurement of the precursor, there are three claims in the literature that are questioned by the results presented here.

First, Banavar and co-workers [218,219] report the observation of terraced precursors for chain molecules similar to those used in this study. Their report of formation of distinct layers is a result of no effort being taken to characterize the precursor. In our simulations, the careful analysis of the precursor shows that the terraced effect is not observed in agreement with findings from Heine et al. [186, 191, 220]. Given that the simulation results of Nieminen et al. [214] were run at temperatures below the melting point, this means that the terraced wetting effect has not yet been captured in an MD study of spreading liquids.

Second, Heine et al. [220] examine the effect of the droplet volume on the spreading dynamics of the precursor and concluded that the diffusion coefficient follows a relation $D_p \propto R_0^x$, where R_0 is the initial radius of the droplet in their simulation setup. They report values of $x = 0.5$ and $x = 0.65$ for two different systems. These findings were based on the incorrect measure of the precursor length r_i, and are therefore possibly flawed. In fact, that the reservoir size influences the diffusive mass transport of the precursor is implausible and is in conflict with any theoretical model on precursor formation described above.

Finally, Albrecht et al. [202] have reported spreading exponents for precursor evolution that are distinctly below the diffusive value 0.5. Their findings, however, are based on the incorrect measure r_i of the precursor length. As shown above, using this incorrect measure can provide spreading exponents that deviate strongly from 0.5 even when the correct width of the precursor l_i has diffusive behavior. Apparently their finding of small spreading exponents is a result of the incorrect measure of the precursor length in combination with the small size of their droplets, which was only a few picoliters. For larger droplets and longer experimental observations, the final state of the precursor is so much larger than the macroscopic shape of the droplet that the effect of using the incorrect measure does not play a dominant role.

4.3 Requirements for precursor formation

The requirement for the formation of precursors are examined in this section. In particular, we examine the role of the spreading coefficient, because it was frequently argued that the onset of precursor formation matches the transition from the partial to the complete wetting regime, corresponding to a change in the sign of the spreading coefficient S [186, 191, 216, 222–224]. It is noteworthy, though, that despite this claim, the spreading coefficient has, to thee best of our knowledge, only been computed in simulations of precursor formation by Wu et al. [222], possibly because its determination is nontrivial. Moreover, we examine the changes of free energy of layer formation, which has not been computed before.

4.3.1 The spreading coefficient

The spreading coefficient S is defined via Equation (1.2) as a force balance of the solid, solid–liquid, and liquid surface energies γ_s, γ_{sl}, and γ_l. While γ_l can be computed easily from MD simulations, γ_s and γ_{sl} cannot be readily accessed.

γ_l is computed from slab geometry simulations of the liquids with the surface of the slab being perpendicular to the z dimension with 16 000 fluid particles in a $23.3\sigma \times 22.5\sigma \times 60.0\sigma$ box for both the atomistic fluid and the chain molecules similar to the setup in Figure 4.4. Simulations were run with a Nosé-Hoover thermostat with damping factor $\tau_d = 0.5\tau$. After equilibration for 500τ, surface tensions were computed from a subsequent run over $20\,000\tau$ using the approach in Appendix A. The remaining simulation settings were similar to the spreading simulations.

The difference between γ_s and γ_{sl} was determined using the thermodynamic integration approach of Leroy et al. [230], in which the potential between the substrate and the liquid is gradually switched off. This is realized in our simulations by scaling ϵ_{sl} for the chain molecule, and ϵ_{sl} and D_{sl} for the atomistic fluid. Simulations were executed with 16 000 fluid particles on a solid substrate with 3 736 particles in a $23.3\sigma \times 22.5\sigma \times 86.0\sigma$ box, in which there is an eight-layer solid slab, a liquid slab, and a vapor slab, as shown in the left image of Figure 4.19. The path was sampled along values of ϵ_{sl} between 0.06ϵ and 3.0ϵ with spacing 0.02ϵ below $\epsilon_{sl} = 2.0\epsilon$ and 0.05 above this threshold. For $\epsilon_{sl} \geq \epsilon$, $D_{sl} = 0.03\epsilon\sigma^{12}$ for the atomistic fluid, in agreement with spreading simulations. Below, D_{sl} is scaled linearly to zero with the same factor as ϵ_{sl}. For $\epsilon_{sl} = 0$, the contribution to the change in free energy is taken as zero [230]. At each ϵ_{sl} the simulations are equilibrated for $500\,\tau$. Production runs are performed for 2000τ during which the change of the potential energy with respect to the change in ϵ_{sl} and D_{sl} is evaluated every 2.5τ. The temperature was controlled using a Nosé-Hoover thermostat with $\tau_T = 0.5\,\tau$. Other settings were as described above. Using this method a solid–liquid interface is removed and separate solid and liquid interfaces are created. The result from this method is thus

$$\Delta\gamma = \gamma_l + \gamma_s - \gamma_{sl}. \tag{4.6}$$

The spreading coefficient can then be determined from $S = \Delta\gamma - 2\gamma_l$. For the atomistic fluid, crystallization of the droplet material was observed in some of these simulations at $\epsilon_{sl} \geq 1.34\epsilon$. Reported spreading coefficients for $\epsilon_{sl} \geq 1.34\epsilon$ are thus influenced by the crystallization. Because this value is above the onset of precursor formation, the information whether the wetting transition matches the comparison to onset of precursors is thus not influenced by this effect.

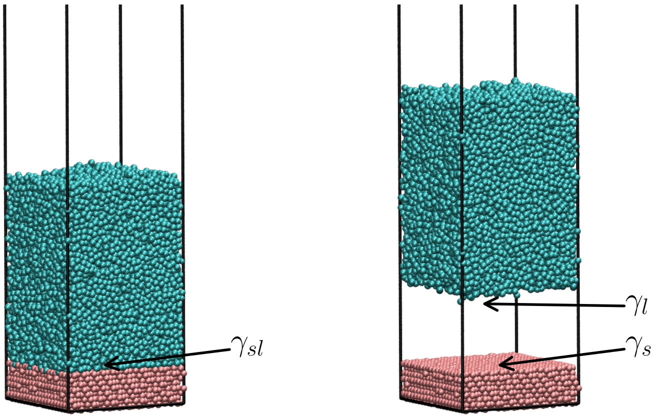


Figure 4.19: Simulation setup for the computation of $\Delta\gamma$ with fluid particles (cyan) and a solid substrate (pink). Box edges are depicted as black lines. Switching off the interactions between the solid and the liquid particles is equivalent to pulling the solid and liquid particles apart and thereby removing the solid–liquid interface and generating a solid and a liquid surface as indicated from the left and right images.

The spreading coefficient as a function of ϵ_{sl} is depicted in Figure 4.20 for the chain molecule and the atomistic fluid. The region in which the transition between no precursor and a precursor appears according to the analysis described in Section 4.2 is shaded. For the chain molecules, the transition from partial wetting to complete wetting matches the onset of the precursor, whereas the onset of a precursor is still well in the partial wetting regime for the atomistic fluid. These results suggest that the onset of precursors can match the transition from the partial to the complete wetting regime, but that it is not a necessity. That the agreement between the onset of precursors and the wetting transition for the chain molecules is not a coincidence will be discussed in the next section.

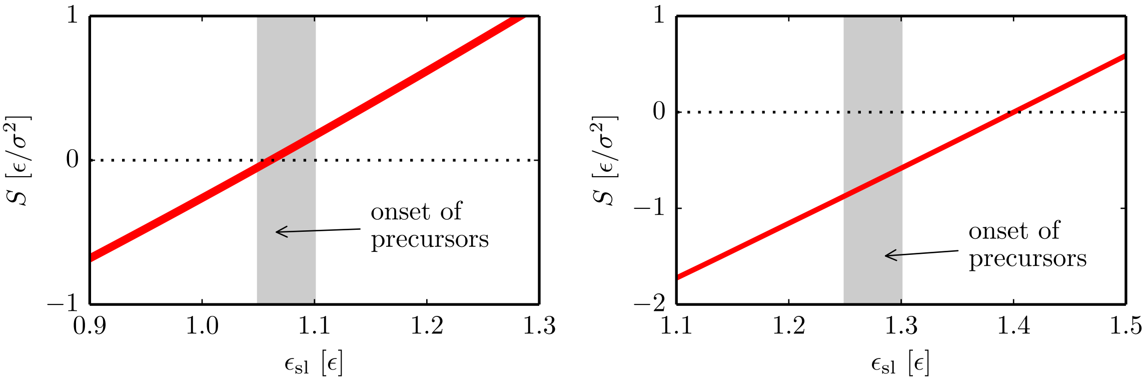


Figure 4.20: Spreading coefficient of the chain molecules (left) and atomistic fluid (right) as a function of the substrate energy. The region in which the onset of precursors occurs is shaded. The width of the red line for the spreading coefficient corresponds to the statistical uncertainty (two-sigma environment) from the simulations.

4.3.2 Free energy of depositing fluid on substrates

Given that processes occur spontaneously if the associated change in free energy is negative, it is logical to assume that precursors will form if the change in free energy of forming a precursor is negative: the formation of layers on the substrate should be energetically favorable. Even though this change in free energy is in principle accessible from molecular simulations, it has not been previously computed. Here we approximate the free energy of forming precursors by computing the change in free energy required to deposit a layer of liquid from a bulk reservoir on a bare solid substrate. In this approach, the deposited layer is a substitute for the precursor and the bulk reservoir is representative for the droplet. This approximation resembles the assumptions that underlie the model of Burlatsky et al. [206, 207].

The changes in free energy are computed with the two-phase thermodynamic (2PT) method [231–233] using the implementation of Lin et al. [234]. This method splits the densities of states, which can be computed from the velocity autocorrelation function, into solid and gas contributions (hence the name of two phases). For each contribution, thermodynamic quantities can be accessed directly using either a harmonic oscillator for the solid component, or the ideal gas approximation for the gas component. Compared to alternative methods to compute the free energy, such as thermodynamic integration or test-particle insertion, which are both exact within statistical uncertainties, this method is an approximation. Its accuracy, however, has been demonstrated for various test cases and was found to be especially striking for single-centered Lennard-Jones particles [231]. Given that the atomistic system studied here resembles single-centered Lennard-Jones particles, we expect the method to provide accurate results for this system. The method has not yet been applied to melts of oligomers or polymers. Its applicability to chain molecules is therefore not warranted. Given the challenges of computing the free energies of chain molecules in solution with the more established free energy methods mentioned above, the approximation with the 2PT method for the chains is still the only viable approach for the systems studied here.

The free energy of the liquid state is computed from bulk systems of fluid. For the atomistic fluid 1 024 particles are randomly positioned in a cubic box of length 12σ. The system is then equilibrated in the NPT ensemble at zero pressure and temperature $T = 0.6\epsilon/k_B$ for 250τ using Nosé-Hoover barostats and thermostats with $\tau_T = 0.5\tau$ and $\tau_p = 5\tau$. Afterwards, the simulation was run for $500\,\tau$ in the NVT ensemble. A Verlet integrator with a timestep of $0.005\,\tau$ was used. Every 50τ, the particle velocities for computing the velocity autocorrelation functions for the 2PT method were dumped with frequency 0.001τ for a period of 10τ.

For the chain molecule 400 molecules were randomly positioned in a cubic box of length 25σ using PACKMOL [71]. The system was then equilibrated in the NPT ensemble at zero pressure using Nosé-Hoover barostats and thermostats with $\tau_T = 0.5\tau$ and $\tau_p = 5\tau$. During the equilibration, the temperature was first slowly increased from 1.2 to $2.0\,\epsilon/k_B$, then run at constant temperature, then slowly cooled down to $1.0\,\epsilon/k_B$ and then ran at $1.0\,\epsilon/k_B$, each of the four steps taking 2000τ. This extended equilibration was used to avoid the system being in a kinetic trap. Afterwards, the system was switched to NVT and run for 2000τ at a temperature of $1.0\,\epsilon/k_B$. Particle velocities were dumped every 200τ with a frequency of 0.001τ for a period of 10τ to compute the velocity-autocorrelation function.

The free energy of the (fluid-laden) substrate was computed from a simulation cell with an eight-layer solid slab consisting of 3736 atoms covered with liquid in a $23.3\sigma \times 22.5\sigma \times 46\sigma$ box. The simulation setup is depicted in Figure 4.21. Simulations were performed for different values of the ratio ρ_{surf} of fluid particles to the surface area of the substrate, with the number of liquid particles varying from 0 to 1200.

For the atomistic system, we use $\epsilon_{sl} = 1.3\epsilon$. A Nosé-Hoover thermostat with damping

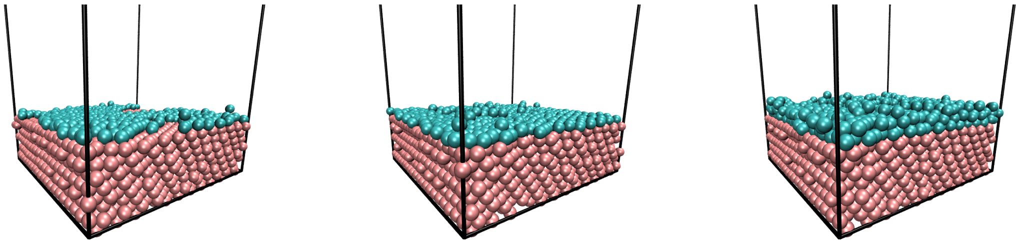


Figure 4.21: Solid substrates (pink) covered with thin layers of liquid (cyan) used in simulations to compute the free energy with the 2PT method.

factor $\tau_T = 0.5\tau$ and a Verlet integrator with time step 0.005τ was used to equilibrate the system in the NVT ensemble for 250τ. Afterwards, simulations were run for 1000τ, with particle velocities dumped every 50τ for 10τ with a frequency of 0.001τ to compute the velocity-autocorrelation functions.

For the chain molecule we use $\epsilon_{sl} = 1.0\epsilon$. The equilibration is performed in the NVT ensemble in which the substrate and the fluid layer are coupled to Nosé-Hoover thermostats separately, both with a damping factor $\tau_T = 0.5\tau$. The temperature of the solid is kept at $1.0\,\epsilon/k_B$ during the equilibration. The temperature of the fluid was raised from 1.0 to $1.5\,\epsilon/k_B$, kept at this temperature, then cooled back to $1.0\,\epsilon/k_B$ and kept at that temperature, each for 250τ. The simulation is then continued for 1000τ during which the particle velocities are dumped every 50τ for 10τ with a frequency of 0.001τ to compute the velocity autocorrelation functions. A Verlet integrator with a timestep of 0.005τ is used in these simulations.

Velocity-autocorrelation functions and the derived thermodynamic quantities were computed separately for each set of particle velocities. Mean values and statistical uncertainties are computed from the results obtained for each separate velocity autocorrelation function. The internal energy is taken directly from the simulations, while the entropy is obtained from the 2PT method. The specific volume of the liquid on the covered substrate, which is required as an input for the 2PT method, was assumed equal to that of the pure liquid. Planck's constant in reduced units was chosen as $h = 0.185\,\sigma(m\epsilon)^{1/2}$ in agreement with [231].

The change in free energy with varying substrate energy compared to the reference energies of $\epsilon_{sl} = 1.3\epsilon$ for the atomistic case and $\epsilon_{sl} = 1.0\epsilon$ for the chain molecules was computed using the thermodynamic integration scheme already used for computing the spreading coefficient. Differences compared to the description above are that for the atomistic fluid we used a Langevin thermostat instead of the Nosé-Hoover thermostat to enhance the sampling, that the equilibration was done for 125τ and the production for 250τ, that a Verlet integrator with a timestep of 0.005τ was used, and that the derivative of the potential was evaluated every 0.625τ. For the chain molecule we used a Nosé-Hoover thermostat and used the same equilibration routines as used in the computations for the 2PT method. After the equilibration the simulation was run for 250τ during which the derivative of the potential energy was evaluated every 0.5τ.

The change in free energy ΔA for depositing fluid layers of the atomistic fluid on the substrate depicted in Figure 4.22 shows characteristic features connected to spreading onset and the different observed precursor types. For low substrate energies, the free energy has a minimum that drops below zero for the first time at $\epsilon_{sl} \approx 1.3\epsilon$, in agreement with the onset of a single-layer precursor at this ϵ_{sl}. Moreover, the minimum is located at a surface density of $\rho_{\text{surf}} \approx 1.0/\sigma^2$, equivalent to a substrate covered by a fluid monolayer. With increasing substrate energy, the minimum at $\rho_{\text{surf}} \approx 1.0/\sigma^2$ is followed by a maximum best visible for $\epsilon_{sl} \approx 1.7\epsilon$, and completely vanishes for the highest substrate energies. While for $\epsilon_{sl} = 1.6\epsilon$ the minimum at $\rho_{\text{surf}} \approx 1.0/\sigma^2$ is a global minimum in the direction of the surface density, ΔA drops below the minimum for $\epsilon_{sl} = 1.8\epsilon$ at $\rho_{\text{surf}} \gtrapprox 1.5/\sigma^2$, in agreement with the transition from a single-layer precursor to a continuously growing precursor in the spreading simulations.

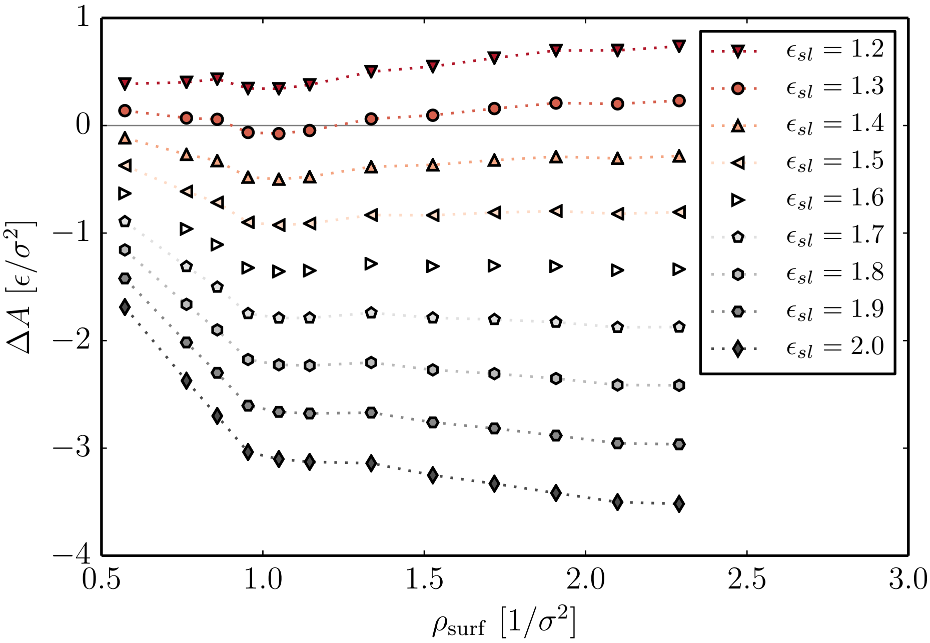


Figure 4.22: Change in free energy of depositing layers of the atomistic fluid from a bulk reservoir on a bare substrate over the surface density ρ_{surf} and different values of ϵ_{sl}. The symbols are larger than the statistical uncertainties (two-sigma environment).

Results for the change in the free energy for the chain molecules are depicted in Figure 4.23. For this system, there is no agreement between a transition to negative free energies and the onset of precursors. It has to be kept in mind, however, that the changes of the free energy shown in this figure are the difference of large values: the free energy of the pure liquid plus the free energy of the bare substrate minus the free energy of the covered substrate. Given that the 2PT method is only an approximation, it is unclear whether it provides sufficiently accurate results for polymer melts in the context of this study; small inaccuracies in, for example, the pure liquid, can result in strong changes in the final results. Decreasing the free energy of the bulk polymer by only a few ϵ, which provides the results given in Figure 4.24, leads to perfect agreement between the change in free energy and the onset of precursors observed at $1.05\epsilon < \epsilon_{sl,c1} < 1.1\epsilon$. We note that we do not argue that there is a specific reason that the bulk system's free energy should be lowered by exactly this value compared to our simulation results, but that an error of only less than 0.5ϵ per bead — which is certainly possible given that the 2PT method is an approximation for the free energy and also that the proper simulation of oligomer melts is more challenging than simulations of the atomistic fluid — can explain discrepancies between the computed free energies and the observations from the spreading simulations.

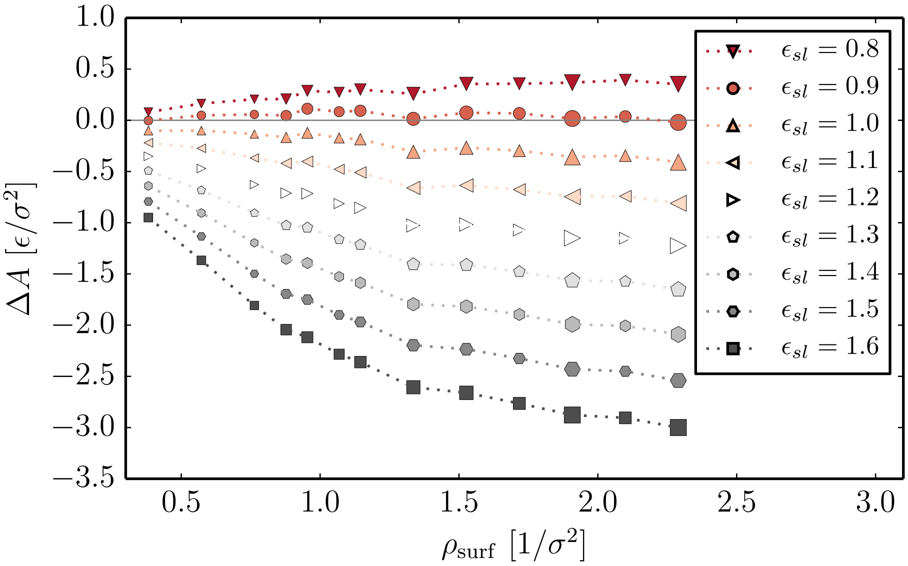


Figure 4.23: Change in free energy of depositing layers of the chain molecules from a bulk reservoir on a bare substrate over the surface density ρ_{surf} and different values of ϵ_{sl}. The symbols' size is equal to statistical uncertainties (two-sigma environment).

Finally, we conclude this section by providing information that resolves the discussion on the conflict of the role of the spreading coefficient for the onset of precursor formation. As shown in Figure 4.25, with increasing ρ_{surf} a layer of increasing thickness is deposited on the substrate. In the limit of $\rho_{\text{surf}} \to \infty$, the deposited layer is so thick

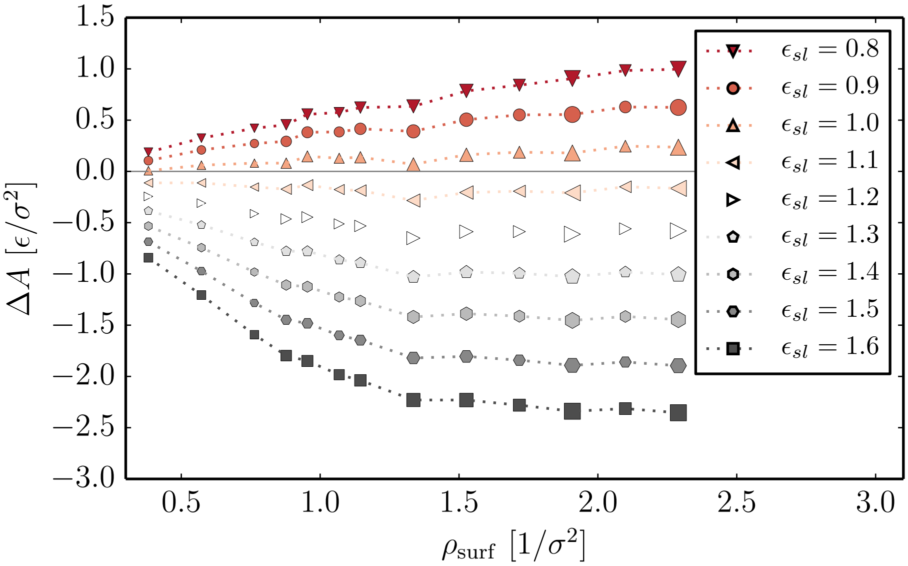


Figure 4.24: Same as Figure 4.23 except that the free energy of the pure liquid is increased by approximately 5ϵ per chain molecule (0.4ϵ per bead).

that the substrate–liquid interface and the liquid–vapor interface do not interfere. The change in free energy of creating this system from a bulk liquid and a bare substrate corresponds to the removal of a solid surface and the creation of solid–liquid and liquid interfaces: $-\gamma_s + \gamma_{ls} + \gamma_l = -S$. Thus, for $\rho_{\mathrm{surf}} \to \infty$, $\Delta A \to -S$. Because $\Delta A = 0$ for $\rho_{\mathrm{surf}} = 0$, the change in the nonwetting–wetting transition has to match the onset of precursors if ΔA does not show a minimum in the direction of ρ_{surf} for a studied system; in this case ΔA cannot drop below 0 in the partial wetting regime and has to drop below 0 in the complete wetting regime. This is in agreement with the results obtained for the chain molecules, where the wetting-nonwetting transition and the onset of the precursor match and a monotonic behavior of ΔA as a function of ρ_{surf} was observed. This finding also implies that a precursor in the partial wetting regime can only exist if ΔA has a minimum in the direction of ρ_{surf} that drops below zero and that a precursor has to exist in the complete-wetting regime for simple fluids.

The examination of ΔA as a function of ρ_{surf} might also explain why the terraced wetting effect has hitherto never been captured properly in MD simulations. Given that the formation of a single layer precursor is described by a shape of ΔA over ρ_{surf} with a single minimum and that a continuously growing precursor is described by a continuously decreasing ΔA, it is plausible to assume that for the terraced effect a curve with several minima, each corresponding to one layer, is to be expected. This is, however, speculation and requires further investigation.

Before concluding this chapter we'd briefly like to comment on the role of the PPPM

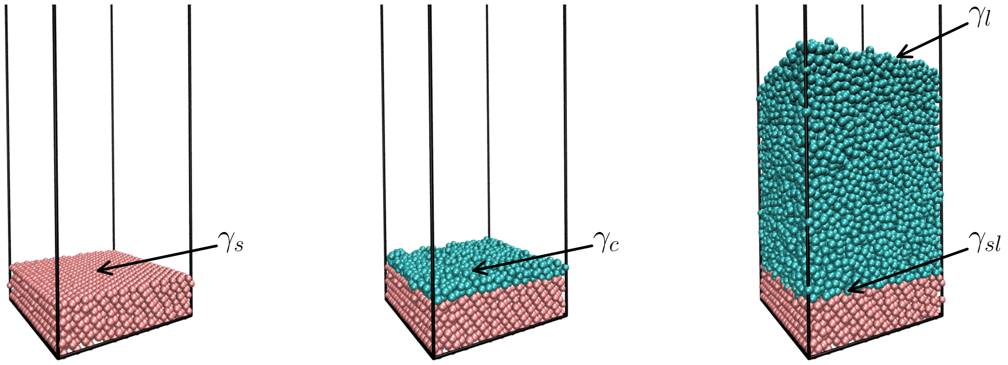


Figure 4.25: Schematic explanation of the role of the spreading coefficient. Pink: solid; cyan: droplet material. The free energy changes when increasing the number of particles on the surface. Starting from the bare substrate with the surface free energy γ_s over a substrate covered with a thin layer of fluid that has an effective surface energy γ_c to a substrate covered with a thick layer. For the thick layer, the upper and the lower interface of the liquid do not interfere and therefor correspond to a liquid and a solid–liquid interface with surface energies γ_l and γ_{sl} and a bulk like phase in between the interfaces. The change of free energy per unit area when moving from the left system to the right system is thus $\gamma_l + \gamma_{sl} - \gamma_s = -S$.

algorithm for the results presented here. The understanding of the requirement for the formation of different kinds of precursors and the role of the spreading coefficient for precursor formation are certainly the most relevant result of the work described in this chapter. The trigger for these findings was mainly given by the minimum that was observed in ΔA for the atomistic fluid in Figure 4.22 and the mismatch of the onset of precursors and the wetting transition in Figure 4.20. Both of these effects are rather subtle. The minimum in ΔA for the relevant substrate energy $\epsilon_{sl} = 1.3\epsilon$ is only approximately $0.2\epsilon/\sigma^2$ below the plateau value at larger values of ρ_{surf}. Likewise, the separation of the onset of precursors and the wetting transition is only $\Delta\epsilon_{sl} \approx 0.1\epsilon$. These subtle energies are smaller than the influence of using a long-range dispersion solver versus truncation, as shown in simulations with the LJ system in Table 2.1,[1] which means that these results might have been missed if the long-range dispersion solvers had not been used. The key findings in this chapter would then be missing.

4.4 Summary

In this chapter we present a series of spreading simulation and novel analysis methods in the context of precursor formation. The superiority of the new analysis method

[1] Note that the uncorrected surface tension γ_p in Table 2.1 is the correct measure for this comparison.

compared to methods used in previous simulation studies is that a clear characterization of precursors is possible. When comparing the new method to previous methods we reveal that previous incorrect measurements of the precursor have led to a series of conclusions that are possibly incorrect, namely: (i) that precursor dynamics can have significant subdiffusive spreading dynamics (cf. Ref. [202]) and (ii) that the effective diffusion coefficient of the precursors is dependent on the size of the droplet (cf. Ref. [220]). Moreover, the analysis suggests that despite many prior claims, the terraced effect has never been captured in simulations of liquid droplets.

In the second part of this chapter we present a series of free energy computations to examine the requirements for the onset of precursor formation and the different types of observed precursors. It is shown that the change in free energy of depositing fluid from a bulk to the substrate correlates with the onset and different types of observed spreading exponents. Using theoretical arguments we also show why the wetting transition can correlate with the onset of precursor formation and thereby resolve the conflict on the role of the spreading coefficient.

5 Summary and future research

Wetting phenomena with strong molecular aspects were examined with the MD simulation method. This work contains the development of a new long-range dispersion solver that was required to accurately model properties of interfacial systems in MD simulations and the study of two selected phenomena, superspreading and molecular precursors.

It was shown that the PPPM Ewald method for dispersion presented in Chapter 2 fulfills all requirements needed for its application in interfacial simulations: long-range forces are accurately captured, the effect on the physical behavior of a system by introducing a cutoff is removed, fast execution, and applicability to large-scale systems. On top of these requirements that motivated the development of the algorithm, it was shown that the transformation of the cutoff from a parameter for tuning the accuracy to a parameter for tuning the efficiency can facilitate simulations that are both more accurate and more efficient than simulations with a simple cutoff. Application of long-range dispersion solvers is thus also beneficial in simulations where their application is not required for an accurate descriptions of the underlying physics.

That computation of long-range dispersion interactions with the PPPM method provides better accuracy, faster simulations, and a closer representation of the true physics of molecular systems suggests that the inclusion of long-range dispersion forces should be the default in MD, just like the long-range treatment of Coulomb interactions is the default nowadays. To achieve this goal, however, many further steps will have to be taken. Because of the global communication required for the 3D FFTs, the PPPM will not perform well in ultra-large scale computations and will possibly also have strong limitations on modern accelerators, namely GPU and Intel Xeon-Phi coprocessors. If long-range dispersion solvers should become the standard, efficient algorithms that allow rapid computation will have to be provided for applications for all system sizes and all hardware architectures. The multilevel summation method or the fast multipole method seem promising candidates to achieve this goal for applications where the PPPM has limitations.

The efficiency and accuracy of PPPM is controlled by various parameters. Their optimal choice is challenging, and despite excessive computations and large parametric studies described here, there is no hint that any of the computations were performed with a set of parameters even close to optimal. Methods that automatically generate optimal parameters to achieve maximum performance while warranting a desired accuracy are thus desirable. The need for such methods will become even more pressing when not only a set of parameters for a method has to be selected, but when the method for treatment of dispersion forces, e.g., continuum corrections, PPPM, MSM, or FMM, is itself an option and therefore drastically increases the parameter space from which the optimum has to be determined.

For the superspreading problem, a molecular mechanism of smoothing the transition at the contact line facilitated by superspreading surfactants has been observed. It was shown that this molecular mechanism complements existing simulation studies on the continuum level and provides plausible explanations for why the superspreading effect occurs and why it occurs only at specific conditions.

Although the proposed mechanism is simple, provides plausible explanations for experimental observations, and is — in contrast to all other mechanisms that have been hypothesized in the literature — based on direct observations, there is no direct evidence provided that the observed mechanism is the true superspreading mechanism. The lack of direct evidence stems from the superspreading phenomenon involving microscopic and macroscopic effects, the latter of which cannot be captured in MD simulations. To provide an incontestable description of the superspreading mechanism by simulation, the effect has to be captured directly, and both microscopic and macroscopic features need to be accessible simultaneously. Simulation methods that might be able to tackle this problem are on the way [235–237], but are still in their infancy. The development of methods capable of capturing an effect as complex as superspreading is possibly still a long way off. Given that these methods will be beneficial not only for superspreading but all phenomena in wetting with both microscopic and macroscopic aspects — which is the case even for the simple spreading of a droplet on a flat substrate — it is certainly worth an investment.

The mechanism of smooth transitions provides an explanation for occurrences at the contact line, but other questions in the context of superspreading remain. In particular, it is unclear how surfactant transport from the interior of the droplet to the surface can work sufficiently fast that the surfactant density at the interfaces does not drop below a threshold where the effect cannot occur. The capacity of trisiloxane surfactants to form vesicles and the vesicles' hypothetical capacity to adsorb directly to the interface might play a deciding role in this context. Coarse-grained MD or other mesoscopic simulation methods are promising tools to examine this problem and may provide answers long before the multiscale methods proposed in the previous paragraph are even close to capturing the superspreading effect.

For molecular precursors, new methods to properly characterize simulation results were developed. These methods and extensive free energy computations provided results that explain the onset of precursor formation, which was the major objective for this part of the study at hand. Nevertheless, not all kinds of precursor shapes known from experiment could be captured in our simulations. In particular, the terraced precursor was not captured adequately. The simulation results at hand suggest, however, that capturing the effect is possible when the system is set up properly and that it is not computationally limited, as has been previously claimed [220]. Future studies could thus attempt to capture and study this effect using suitable molecular models and system setups.

The simulations in the context of molecular precursors were performed with model fluids on smooth, crystalline substrates. In other words, the simulation setup could not have been any simpler. The understanding about the need for proper analysis of these simulations and the physical insight about the onset of precursor formation pave the way for tackling systems with more complex features, such as realistic potentials for the liquid and substrate, multifunctional or amphiphilic molecules, or substrates with nanopatterns, nanoscale roughness, or chemical inhomogeneities.

Appendix

A Surface tensions and liquid densities from slab simulations

Because surface tensions obtained from slab simulations are reported several times in this thesis, their derivation is given here in the appendix to avoid unnecessary repetition. Surface tensions can be obtained from MD simulations via two-phase simulations. We use the approach, developed by Tolman [238] and Kirkwood and Buff [239], in which the surface tension is expressed via

$$\gamma_p = \frac{1}{2} \int_{-\infty}^{\infty} \big(p_\perp(z) - p_\parallel(z)\big)\, \mathrm{d}z, \tag{A.1}$$

where $p_\perp(z) = p_z(z)$ is the pressure component normal to the surface and $p_\parallel(z) = (p_x(z) + p_y(z))/2$ is the pressure component parallel to the surface. Replacing the integral with an ensemble average leads to

$$\gamma_p = \frac{L_z}{2} \left(p_\perp - p_\parallel\right) = \frac{L_z}{2} \left[\langle p_z \rangle - \frac{\langle p_x \rangle + \langle p_y \rangle}{2}\right], \tag{A.2}$$

where L_z is the box dimension in the z-direction. The outer factor of 1/2 takes into account that the simulated system contains two interfaces.

If a cutoff is introduced for the pair potential, the surface tensions calculated with Equation (A.2) will underestimate the correct surface tension of the simulated material. This error can be estimated by adding a “tail correction” γ_{tail} to the simulated surface tension to provide a better estimate of the correct surface tension

$$\gamma \approx \gamma_p + \gamma_{\text{tail}} \tag{A.3}$$

from the simulation. The correction can be calculated as [21, 22]

$$\begin{aligned} \gamma_{\text{tail}} &= \frac{\pi}{2} \int_{-\infty}^{\infty} \int_{-1}^{1} \int_{r_c}^{\infty} r^3 \frac{\mathrm{d}U(r)}{\mathrm{d}r} g(r)(1 - 3s^2) \\ &\quad \times \left(\rho(z)\rho(z - sr) - (\rho_{\mathrm{G}}(z))^2\right) \mathrm{d}r\mathrm{d}s\mathrm{d}z, \end{aligned} \tag{A.4}$$

where $U(r)$ is the pair potential, $g(r)$ is the radial distribution function, $\rho(z)$ is the simulated density profile, r_c is the cutoff radius for the pair potential, and $\rho_{\mathrm{G}}(z)$ is the Gibbs dividing surface

$$\rho_{\mathrm{G}}(z) = \rho_c + \frac{\Delta\rho}{2}\mathrm{sgn}(z), \tag{A.5}$$

where ρ_c is the mean and $\Delta\rho$ is the difference of the densities of the coexisting phases. $g(r)$ was assumed to be unity beyond the cutoff in the calculations of the tail correction. The values for ρ_c and $\Delta\rho$, which were also used to calculate the liquid and vapor densities in this study, were obtained from fitting an error function to the simulated density profile [78, 240–242]. Statistical uncertainties for the surface tensions were obtained using the blocking method [243]. When statistical uncertainties are reported for the density, those were computed from subdividing the trajectory into 5 blocks and determining the liquid density for each of the blocks to compute mean values and standard deviations of the mean.

B Force field parameters

The total potential U_{total} of a standard two-body force field for simulations in Chapter 3 is composed of bonded U_{bonded} and nonbonded $U_{\text{nonbonded}}$ interactions:

$$U_{\text{total}} = U_{\text{bonded}} + U_{\text{nonbonded}}. \tag{B.1}$$

B.1 Nonbonded interactions

The nonbonded interactions consist of terms for Coulomb interactions U_{coul} and for van der Waals (vdW) interactions U_{vdW}:

$$U_{\text{nonbonded}} = U_{\text{coul}} + U_{\text{vdW}}. \tag{B.2}$$

The Coulomb term is described via

$$U_{\text{coul}} = \frac{1}{4\pi\epsilon_0} \sum_{i,j>i} \frac{q_i q_j}{r_{ij}}, \tag{B.3}$$

where ϵ_0 is the vacuum permittivity, the sum is over all pairs of particles in the system, q_i is the fixed point charge of atom i, and r_{ij} is the distance between particles i and j. The vdW term is described either by a Buckingham potential

$$U_{\text{Buck}} = \sum_{i,j>i} A_{ij} \exp(-B_{ij} r_{ij}) - \frac{C_{6,ij}}{r_{ij}^6}, \tag{B.4}$$

where A, B, and C_6 are coefficients to describe the interactions, or a LJ potential

$$U_{\text{LJ}} = \sum_{i,j>i} \frac{C_{12,ij}}{r_{ij}^{12}} - \frac{C_{6,ij}}{r_{ij}^6}, \tag{B.5}$$

where C_{12} is a coefficient. The C_6 coefficients follow geometric mixing rules

$$C_{6,ij} = \sqrt{C_{6,ii} C_{6,jj}}. \tag{B.6}$$

Nonbonded interactions are computed for all atoms of different molecules and for atoms within the same molecule that are separated by more than 2 bonds. Nonbonded interactions of atoms that are separated by 3 bonds are scaled by f_{1-4}. This scaling factor was taken from the original polymer models and is either 1.0 or 0.5. The factor for specific pairs is given in Table B.5.

The coefficients for the nonbonded interactions are given in Tables B.1 and B.2. The charges are given in multiples of the elementary charge e. The subscripts of the atom types indicate to which molecule type an atom belongs; for example, $\mathrm{C_{PDMS}}$ is a carbon atom of a PDMS molecule. The classification of the atom types for the surfactants for vdW interactions is given in Figure B.1.

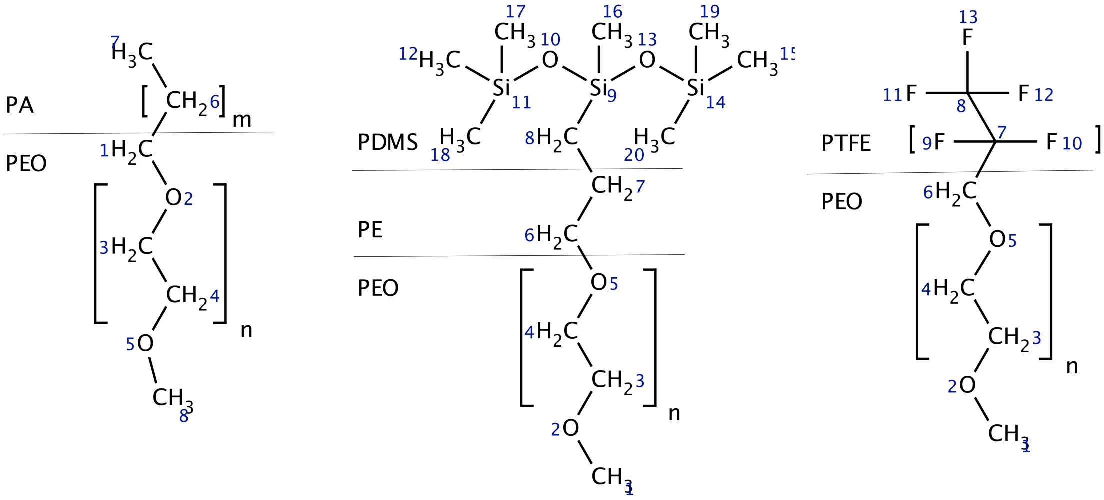


Figure B.1: Surfactants used in this study. The different regions of the surfactants separated by horizontal lines define to which vdW type the atoms are assigned. The blue numbers are used to identify special charges and bonded parameters.

Atom Type	q [e]	C_6 [kcalÅ^6/mol]	reference for q	reference for C_6
Water				
$\mathrm{O_{water}}$	-1.1128	736.085	[119]	[119]
$\mathrm{H_{water}}$	0.5564	0.0	[119]	[119]
PA				
$\mathrm{C_{PA}}$	0.0	660.024	[53]	this work
$\mathrm{H_{PA}}$	0.0	28.222	[53]	this work
PTFE				
$\mathrm{C_{PTFE}}$	0.0	660.024	[54]	this work
$\mathrm{F_{PTFE}}$	0.0	135.72	[54]	this work
PDMS				
$\mathrm{C_{PDMS}}$	-0.5604	640.8	[118]	[118]
$\mathrm{H_{PDMS}}$	0.137	27.4	[118]	[118]
$\mathrm{O_{PDMS}}$	-0.462	398.9	[118]	[118]
$\mathrm{Si_{PDMS}}^a$	0.7608	3085.3	[118]	[118]
$\mathrm{Si_{PDMS}}^b$	0.6792	3085.3	[118]	[118]
PEO				
$\mathrm{C_{PEO}}^a$	-0.0326	650.352	[117]	this work
$\mathrm{C_{PEO}}^b$	-0.1187	650.352	[117]	this work
$\mathrm{H_{PEO}}$	0.0861	27.7644	[117]	this work
$\mathrm{O_{PEO}}$	-0.2792	404.838	[117]	this work
Trisiloxane surfactant^c				
$\mathrm{C_{12,15-20}}$	-0.4275	640.8	[118]	[118]
$\mathrm{C_8}$	-0.4887	640.8	this work	this work
$\mathrm{C_7}$	0.0598	660.024	this work	this work
$\mathrm{C_6}$	0.0277	660.024	this work	this work
$\mathrm{H_{12,15-20}}$	0.0861	27.4	[118]	[118]
$\mathrm{H_8}$	0.0977	27.4	this work	this work
$\mathrm{H_7}$	0.0244	28.222	this work	this work
$\mathrm{H_6}$	0.0621	27.7644	this work	this work
$\mathrm{O_{10,13}}$	-0.5302	398.9	[118]	[118]
$\mathrm{Si_9}$	0.8612	3085.3	[118]	[118]
$\mathrm{Si_{11,14}}$	0.778	3085.3	[118]	[118]

[a] middle group; [b] terminal group; [c] Charges not defined explicitly are PEO specific charges.

Table B.1: Partial charges and dispersion coefficients. Element indices correspond to the blue numbers in Figure B.1.

Atom Type 1[a]	Atom Type 2	A [kcal/mol]	B [$^{-1}$]	C_{12} [kcal^{12}/mol]	reference
C_*	C_*	14976	3.09		[53, 54, 117, 118]
C_*	H_*	4320	3.415		[53, 117]
C_*	O_*	33702	3.577		[53, 117, 118]
C_*	F_*	6923.2	3.084		[54]
C_*	Si_*	24753	2.968		[53, 117, 118]
H_*	H_*	2649	3.74		[53, 117, 118]
H_*	O_*	14176	3.9015		[117, 118]
H_*	F_*	12983.41692	4.0005		this work
H_*	Si_*	11548	3.293		[118]
O_*	O_*	75844	4.063		[117, 118]
O_*	F_*	69471.814	4.162		this work
O_*	Si_*	61781	3.454		[118]
F_*	F_*	63635	4.261		[54]
F_*	Si_*	56590.59118	3.5535		this work
Si_*	Si_*	50326	2.846		[118]
C_{PDMS}	H_{PDMS}	6300	3.415		[118]
O_{water}	C_*			545 888.165 990 791	this work
O_{water}	H_*			97 991.000 487 818 6	this work
O_{water}	O_*			282 583.136 478 903	this work
O_{water}	F_*			339 379.491 056 056	this work
O_{water}	Si_*			726 207.735 099 629	this work
H_{water}	C_*			0.0	this work
H_{water}	H_*			0.0	this work
H_{water}	O_*			0.0	this work
H_{water}	F_*			0.0	this work
H_{water}	Si_*			0.0	this work
O_{water}	O_{water}			731 380.256 787 889	[119]
O_{water}	H_{water}			0.0	[119]
H_{water}	H_{water}			0.0	[119]

[a] Atoms labeled with "$*$" are generic types. Atom labeled with a molecule name are exceptions to the generic coefficients.

Table B.2: vdW coefficients for the repulsive terms.

B.2 Bonded interactions

The geometry of the water molecule is rigid. The bonded interactions of the other molecules are composed of bond, angle, and dihedral potentials U_{bond}, U_{angle}, and U_{dihedral}. The bonded potential between two connected atoms l and m is

$$u_{\mathrm{bond}} = \sum_{i=2}^{4} k_{i,lm}^{\mathrm{bond}} (r_{lm} - r_{lm}^{0})^{i}, \tag{B.7}$$

where r_{lm} is the distance between those particles, r_{lm}^{0} is their equilibrium distance, and $k_{i,lm}^{\mathrm{bond}}$ are coefficients.

The angle potential between three connected atoms l, m, and n is

$$u_{\mathrm{angle}} = \sum_{i=2}^{4} k_{i,lmn}^{\mathrm{angle}} (\theta_{lmn} - \theta_{lmn}^{0})^{i}, \tag{B.8}$$

where θ_{lmn} is the angle between those particles, θ_{lmn}^{0} is the equilibrium angle, and $k_{i,lmn}^{\mathrm{angle}}$ are coefficients.

The dihedral potential between four connected atoms l, m, n, and o is

$$u_{\mathrm{dihedral}} = \frac{1}{2} \sum_{i=1}^{7} k_{i,lmno}^{\mathrm{dihedral}} [1 - \cos(i\phi_{lmno})], \tag{B.9}$$

where $k_{i,lmno}^{\mathrm{dihedral}}$ are coefficients to describe the interactions between particles l, m, n, and o, and ϕ_{lmno} is the dihedral angle between those atoms. An exception is the Si-O-Si-O dihedral potential, which is computed with [118]

$$u_{\mathrm{dihedral,Si,O,Si,O}} = \frac{\sin^{2}(\theta_{\mathrm{Si,O,Si}})}{\sin^{2}(\theta_{\mathrm{Si,O,Si}}^{0})} \times \frac{1}{2} \sum_{i=1}^{7} k_{i,\mathrm{Si,O,Si,O}}^{\mathrm{dihedral}} [1 - \cos(i\phi_{\mathrm{Si,O,Si,O}})]. \tag{B.10}$$

The bonded interaction coefficients are given in Tables B.3, B.4, and B.5.

l	m	$k_{2,lm}^{\text{bond}}$ [kcal/mol/Å^2]	$k_{3,lm}^{\text{bond}}$ [kcal/mol/Å^3]	$k_{4,lm}^{\text{bond}}$ [kcal/mol/Å^4]	r_{lm}^0 [Å]	reference
Water						
O	H				0.9572	[119]
PA						
C	C	309			1.53	[229]
C	H	327			1.1	[229]
PTFE						
C	C[a]	308.9			1.5727	[54]
C	C[b]	308.9			1.5658	[54]
C	F[a]	361			1.351	[54]
C	F[b]	361			1.3391	[54]
PDMS						
Si	O	350	-517	674	1.651	[118]
Si	C	190	-279	308	1.878	[118]
C	H	328			1.092	[118]
PEO						
C	C	309			1.5075	[244]
C	O	369.5			1.4115	[244]
C	H	327.5			1.1041	[244]
alkyl-ethoxylate surfactant						
C_1	C_6	309			1.5075	[117]
trisiloxane surfactant						
C_7	C_8	309			1.5075	[117]
C_7	H_7	327.5			1.1041	[117]
C_6	C_7	309			1.5075	[117]
C_6	H_6	327.5			1.1041	[117]
perfluoroalkane surfactant						
C_7	C_6	308.9			1.517	[54][c]

[a] middle group; [b] terminal group; [c] force constant taken from the literature, equilibrium distance from this work;

Table B.3: Bond coefficients. Special bonded interactions of the surfactants are labeled with numbers corresponding to those in Figure B.1. Bonded parameters not given explicitly for the surfactants correspond to generic types.

			$k^{\text{angle}}_{2,lmn}$	$k^{\text{angle}}_{3,lmn}$	$k^{\text{angle}}_{4,lmn}$	θ^0_{lmn}	
l	m	n	[kcal/mol/rad^2]	[kcal/mol/rad^3]	[kcal/mol/rad^4]	[deg]	reference
Water							
H	O	H				104.52	[119]
PA							
C	C	C	54			114.4	[229]
C	C	H	44			110.5	[229]
H	C	H^a	38.5			108.2	[229]
H	C	H^b	38.5			107.7	[229]
PTFE							
C	C	C	80.3			115.55	[54]
C	C	F	90			109.46	[54]
F	C	F^a	120			110.13	[54]
F	C	F^b	120			108.54	[54]
PDMS							
Si	O	Si	10.305	-18.101	10.1	137.63	[118]
O	Si	O	91.835			105.56	[118]
O	Si	C	23.022	-31.399	24.981	109.82	[118]
C	Si	C	36.21	-20.39	20.02	112.44	[118]
Si	C	H	28.77	-13.95		111.09	[118]
H	C	H	38.5			107.77	[118]
PEO							
C	C	H	42.9			110.1	[117]
H	C	H	38.5			109.47	[117]
O	C	H	56			109.48	[117]
C	O	C	74.5			108.05	[117]
O	C	C	86			108.54	[117]
alkyl-ethoxylate surfactant							
C$_6$	C$_6$	C$_1$	54			114.4	[229]
O$_2$	C$_1$	C$_6$	86			108.54	[117]
C$_2$	C$_6$	H$_6$	44			110.5	[229]
C$_6$	C$_2$	H$_2$	44			110.5	[229]
trisiloxane surfactant							
Si$_9$	C$_8$	C$_7$	46.45			115.35	this work
C$_8$	C$_7$	C$_6$	54			114.4	[229]
C$_7$	C$_8$	H$_8$	42.9			110.1	[117]
C$_8$	C$_7$	H$_7$	42.9			110.1	[117]
C$_7$	C$_6$	H$_6$	42.9			110.1	[117]
C$_6$	C$_7$	H$_7$	42.9			110.1	[117]
perfluoroalkane surfactant							
C$_7$	C$_7$	C$_6$	44.16			115.6	this work
F$_7$	C$_7$	C$_6$	90			110.3	[54]^c
C$_7$	C$_6$	H$_6$	42.9			108.71	[117]^c
C$_7$	C$_6$	O$_5$	95.6			107.1	this work

^{a} middle group ^{b} terminal group ^{c} force constant taken from the literature, equilibrium angle from this work;

Table B.4: Angle coefficients. Special bonded interactions of the surfactants are labeled with numbers corresponding to those in Figure B.1. Bonded parameters not given explicitly for the surfactants correspond to generic types.

				$k_{i,lmno}^{\text{dihedral}}$ [kcal/mol]								
l	m	n	o	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	f_{1-4}	reference
PA												
C	C	C	C	0.06	0.132	0.06	-0.429				1.0	[229]
C	C	C	H			-0.3					1.0	[229]
H	C	C	H			-0.3					1.0	[229]
PTFE												
C	C	C	C	-0.925	0.07	1.427	-0.54	-0.207		0.676	1.0	[54]
C	C	C	F								1.0	[54]
F	C	C	F			-0.38					1.0	[54]
PDMS												
Si	O	Si	O	-0.41	0.27319	0.21968					0.5	[118]
Si	O	Si	C			-0.05706					0.5	[118]
O	Si	C	H			-0.15					0.5	[118]
C	Si	C	H			-0.15					0.5	[118]
PEO												
O	C	C	H			-0.28					1.0	[117]
H	C	C	H			-0.28					1.0	[117]
C	O	C	H			-0.73					1.0	[117]
O	C	C	O	0.47	-2.43	-0.36	-0.95	-0.45			1.0	[117]
C	O	C	C	1.87	1.17	0.46	-0.37				1.0	[117]
alkyl-ethoxylate surfactant												
C_6	C_6	C_6	C_1	0.06	0.132	0.06	-0.429				1.0	[229]
O_2	C_1	C_6	C_6	2.56	0.33	-1.04					1.0	this work
H_1	C_1	C_6	H_6			-0.28					1.0	[117]
trisiloxane surfactant												
$O_{10,13}$	Si_9	C_8	C_7			-0.19					0.5	this work
C_{16}	Si_9	C_8	C_7			-0.19					0.5	this work
Si_9	C_8	C_7	H_7			-0.11					0.5	this work
Si_9	C_8	C_7	C_6			-0.11					0.5	this work
C_8	C_7	C_6	O_5	2.56	0.33	-1.04					1.0	this work
H_8	C_8	C_7	H_7			-0.28					1.0	[117]
H_6	C_6	C_7	H_7			-0.28					1.0	[117]
perfluoroalkane surfactant												
C_7	C_7	C_7	C_6	-0.791	-0.47	-2.03					1.0	this work
F_7	C_7	C_7	C_6	-1.686	-1.650	-0.991					1.0	this work
C_7	C_7	C_6	H_6	-0.073	0.129	-0.157					1.0	this work
C_7	C_7	C_6	O_5	-0.071	-0.137	0.079					1.0	this work
F_7	C_7	C_6	H_6	0.073	-0.122	-0.334					1.0	this work
F_7	C_7	C_6	O_5	-0.071	0.13	-0.167					1.0	this work
C_7	C_6	O_5	C_4	-0.422	-0.513	2.113					1.0	this work

Table B.5: Dihedral coefficients and f_{1-4} scaling factors. The scaling factor is for the first and last atom of the dihedral. Special bonded interactions of the surfactants are labeled with numbers corresponding to those in Figure B.1. Bonded parameters not given explicitly for the surfactants correspond to generic types.

Bibliography

[1] P.-G. De Gennes, F. Brochard-Wyart, and D. Quéré. *Capillarity and Wetting Phenomena*. Springer Verlag, 2004.

[2] W. Hild, A. Opitz, J. A. Schaefer, and M. Scherge. The effect of wetting on the microhydrodynamics of surfaces lubricated with water and oil. *Wear*, 254:871–875, 2003.

[3] R. M. Hill. Superspreading. *Current Opinion in Colloid & Interface Science*, 3:247–254, 1998.

[4] T. Kraus, L. Malaquin, H. Schmid, W. Riess, N. D. Spencer, and H. Wolf. Nanoparticle printing with single-particle resolution. *Nature Nanotechnology*, 2:570–6, 2007.

[5] N. R. Morrow. Wettability and Its Effect on Oil Recovery. *Journal of Petroleum Technology*, 42:1476–1484, 1990.

[6] R. Fürstner, W. Barthlott, C. Neinhuis, and P. Walzel. Wetting and self-cleaning properties of artificial superhydrophobic surfaces. *Langmuir*, 21:956–61, 2005.

[7] P. M. Kistler, S. F .and Schweizer. *Liquid Film Coating*. Springer, Berlin, Heidelberg, 1997.

[8] T. Young. An essay on the cohesion of fluids. *Philosophical Transactions of the Royal Society of London*, 95:65, 1805.

[9] N. K. Adam. Use of the Term “Young’s Equation” for Contact Angles. *Nature*, 180:809–810, 1957.

[10] D. Bonn, J. Eggers, J. Indekeu, J. Meunier, and E. Rolley. Wetting and spreading. *Reviews of Modern Physics*, 81:739–805, 2009.

[11] P.-G. De Gennes. Wetting: statics and dynamics. *Reviews of Modern Physics*, 57:827–863, 1985.

[12] C. Huh and L. E. Scriven. Hydrodynamic model of steady movement of a solid/liquid/fluid contact line. *Journal of Colloid and Interface Science*, 35:85–101, 1971.

[13] H. Brenner and V. Ganesan. Molecular wall effects: Are conditions at a boundary "boundary conditions"? *Physical Review E*, 61:6879–6897, 2000.

[14] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids.* Oxford University Press, Oxford, 1987.

[15] J. N. Israelachvili. *Intermolecular and Surface Forces.* Elsevier, third edition, 2011.

[16] M. N. Popescu, G. Oshanin, S. Dietrich, and A.-M. Cazabat. Precursor films in wetting phenomena. *Journal of Physics: Condensed Matter*, 24:243102, 2012.

[17] L. H. Tanner. The spreading of silicone oil droplets on horizontal surfaces. *Journal of Physics D: Applied Physics*, 12:1473, 1979.

[18] J. Venzmer. Superspreading - 20 years of physiochemical research. *Current Opinion in Colloid & Interface Science*, 16:335–343, 2011.

[19] F. Heslot, N. Fraysse, and A.-M. Cazabat. Molecular layering in the spreading of wetting liquid drops. *Nature*, 338:640–642, 1989.

[20] F. Heslot, A.-M. Cazabat, and P. Levinson. Dynamics of wetting of tiny drops: Ellipsometric study of the late stages of spreading. *Physical Review Letters*, 62:1286–1289, 1989.

[21] G. A. Chapela, G. Saville, S. M. Thompson, and J. S. Rowlinson. Computer simulation of the gas/liquid surface. *Journal of the Chemical Society - Faraday Transactions II*, 73:1133–1144, 1977.

[22] E. M. Blokhuis, D. Bedeaux, C. D. Holcomb, and J. A. Zollweg. Tail correction to the surface tension of a lennard-jones liquid-vapour interface. *Molecular Physics*, 85:665–669, 1995.

[23] P. J. in 't Veld, A. E. Ismail, and G. S. Grest. Application of Ewald summations to long-range dispersion forces. *Journal of Chemical Physics*, 127:144711, 2007.

[24] J. Janeček. Long range corrections in inhomogeneous simulations. *Journal of Physical Chemistry B*, 110:6264–6269, 2006.

[25] S. Werth, G. Rutkai, J. Vrabec, M. Horsch, and H. Hasse. Long-range correction for multi-site Lennard-Jones models and planar interfaces. *Molecular Physics*, 0:1–8, 2013.

[26] R. A. Zubillaga, A. Labastida, B. Cruz, J. C. Martínez, E. Sánchez, and J. Alejandre. Surface Tension of Organic Liquids Using the OPLS/AA Force Field. *Journal of Chemical Theory and Computation*, 9:1611–1615, 2013.

[27] W. J. Jorgensen, D. S. Maxwell, and J. Tirado-Rives. Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids. *Journal of the American Chemical Society*, 118:11225–11236, 1996.

[28] P. Lagüe, R. W. Pastor, and B. R. Brooks. Pressure-Based Long-Range Correction for Lennard-Jones Interactions in Molecular Dynamics Simulations: Application to Alkanes and Interfaces. *Journal of Physical Chemistry B*, 108:363–368, 2004.

[29] H. Wang, C. Schütte, and P. Zhang. Error estimate of short-range force calculation in inhomogeneous molecular systems. *Physical Review E*, 86:026704, 2012.

[30] P. Springer. *A Scalable, Linear-Time Dynamic Cutoff Algorithm for Molecular Simulations of Interfacial Systems.* Masterthesis, RWTH Aachen University, Germany, 2014.

[31] V. Rokhlin. Rapid solution of integral equations of classical potential theory. *Journal of Computational Physics*, 60:187–207, 1985.

[32] R. D. Skeel, I. Tezcan, and D. J. Hardy. Multiple Grid Methods for Classical Molecular Dynamics. *Journal of Computational Chemistry*, 23:673–684, 2002.

[33] P. P. Ewald. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Annalen der Physik*, 369:253–287, 1921.

[34] R. Hockney and J. Eastwood. *Computer simulations using particles.* McGraw-Hill Inc., New York, 1988.

[35] T. Darden, D. York, and L. Pederson. Particle mesh Ewald: An N log(N) method for Ewald sums in large systems. *Journal of Chemical Physics*, 98:10089–10092, 1993.

[36] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen. A smooth particle mesh Ewald method. *Journal of Chemical Physics*, 103:8577–8593, 1995.

[37] A. Arnold, F. Fahrenberger, C. Holm, O. Lenz, M. Bolten, H. Dachsel, R. Halver, I. Kabadshow, F. Gähler, F. Heber, J. Iseringhausen, M. Hofmann, M. Pippig, D. Potts, and G. Sutmann. Comparison of scalable fast methods for long-range interactions. *Physical Review E*, 063308, 2013.

[38] D. Tameling, P. Springer, P. Bientinesi, and A. E. Ismail. Multilevel summation for dispersion: A linear-time algorithm for r^{-6} potentials. *Journal of Chemical Physics*, 140:024105, 2014.

[39] K. Lorenzen, C. Wichmann, and P. Tavan. Including the Dispersion Attraction into Structure-Adapted Fast Multipole Expansions for MD Simulations. *Journal of Chemical Theory and Computation*, 10:3244–3259, 2014.

[40] D. E. Williams. Accelerated Convergence of Crystal-Lattice Potential Sums. *Acta Crystallograrphica Section A*, 27:452–455, 1971.

[41] J. W. Perram, H. G. Petersen, and S. W. De Leeuw. An algorithm for the simulation of condensed matter which grows as the 3/2 power of the number of particles. *Molecular Physics*, 65:875–893, 1988.

[42] N. Karasawa and W. A. Goddard, III. Acceleration of Convergence for Lattice Sums. *Journal of Physical Chemistry*, 93:7320–7327, 1989.

[43] M. Mecke, J. Winkelmann, and J. Fischer. Molecular dynamics simulation of the liquid-vapor interface: The Lennard-Jones fluid. *Journal of Chemical Physics*, 107:9264–9270, 1997.

[44] J. López-Lemus, M. Romero-Bastida, T. A. Darden, and J. Alejandre. Liquid-vapour equilibrium of n-alkanes using interface simulations. *Molecular Physics*, 104:2413–2421, 2006.

[45] A. E. Ismail, M. Tsige, P. J. in 't Veld, and G. S. Grest. Surface tension of normal and branched alkanes. *Molecular Physics*, 105:3155–3163, 2007.

[46] W.-Y. Ou-Yang, Z.-Y. Lu, T.-F. Shi, Z.-Y. Sun, and L.-J. An. A molecular-dynamics simulation study on the dependence of Lennard-Jones gas-liquid phase diagram on the long-range part of the interactions. *Journal of Chemical Physics*, 123:234502, 2005.

[47] J. Alejandre and G. A. Chapela. The surface tension of TIP4P/2005 water model using the Ewald sums for the dispersion interactions. *Journal of Chemical Physics*, 132:014701, 2010.

[48] C. L. Wennberg, T. Murola, B. Hess, and E. Lindahl. Lennard-Jones Lattice Summation in Bilayer Simulations Has Critical Effects on Surface Tension and Lipid Properties. *Journal of Chemical Theory and Computation*, 9:3527–3537, 2013.

[49] B. Shi, S. Sinha, and V. K. Dhir. Molecular Dynamics simulation of the density and surface tension of water by particle-particle particle-mesh method. *Journal of Chemical Physics*, 124:204715, 2006.

[50] M. Deserno and C. Holm. How to mesh up Ewald sums. I. A theoretical and numerical comparison of various particle mesh routines. *Journal of Chemical Physics*, 109:7678–7693, 1998.

[51] D. M. Heyes. Electrostatic potentials and fields in infinite point charge lattices. *Journal of Chemical Physics*, 74:1924, 1981.

[52] B. R. A. Nijboer and F. W. De Wette. On the calculation of lattice sums. *Physica*, 23:309–321, 1957.

[53] R. A. Sorensen, W. B. Liau, L. Kesner, and R. H. Boyd. Prediction of Polymer Crystal Structures and Properties. Polyethylene and Poly(oxymethylene). *Macromolecules*, 21:200–208, 1988.

[54] O. Borodin, G. D. Smith, and D. Bedrov. A Quantum Chemistry Based Force Field for Perfluoroalkanes and Poly(tetrafluoroethylene). *Journal of Physical Chemistry B*, 106:9912–9922, 2002.

[55] L. Zhang and J. I. Siepmann. Pressure Dependence of the Vapor-Liquid-Liquid Phase Behavior in Ternary Mixtures Consisting of n-Alkanes, n-Perfluoroalkanes, and Carbon Dioxide. *Journal of Physical Chemistry B*, 109:2911–2919, 2005.

[56] T. Schnabel, A. Srivastava, J. Vrabec, and H. Hasse. Hydrogen Bonding of Methanol in Supercritical CO2: Comparison between 1H NMR Spectroscopic Data and Molecular Simulation Results. *Journal of Physical Chemistry B*, 111:9871–9878, 2007.

[57] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19:297–301, 1965.

[58] B. A. Luty, I. G. Tironi, and W. F. van Gunsteren. Lattice-sum methods for calculating electrostatic interactions in molecular simulations. *Journal of Chemical Physics*, 103:3014–3021, 1995.

[59] P. H. Hünenberger. Lattice-sum methods for computing electrostatic interactions in molecular simulations. *AIP Conference Proceedings*, 492:17–83, 1999.

[60] V. Ballenegger, J. J. Cerdà, and C. Holm. How to Convert SPME to P3M: Influence Functions and Error Estimates. *Journal of Chemical Theory and Computation*, 8:936–947, 2012.

[61] H. A. Stern and K. G. Calkins. On mesh-based Ewald methods: optimal parameters for two differentiation schemes. *Journal of Chemical Physics*, 128:214106, 2008.

[62] V. Ballenegger, J. J. Cerdà, and C. Holm. Removal of spurious self-interactions in particle-mesh methods. *Computer Physics Communications*, 182:1919–1923, 2011.

[63] A. Neelov and C. Holm. Interlaced P3M algorithm with analytical and ik-differentiation. *Journal of Chemical Physics*, 132:234103, 2010.

[64] V. Ballenegger, J. J. Cerda, O. Lenz, and C. Holm. The optimal P3M algorithm for computing electrostatic energies in periodic systems. *Journal of Chemical Physics*, 128:034109, 2008.

[65] J. Kolafa and J. W. Perram. Cutoff errors in the Ewald summation formulae for point charge systems. *Molecular Simulation*, 9:351–368, 1992.

[66] M. Deserno and C. Holm. How to mesh up Ewald sums. II. An accurate error estimate for the P^{3}M algorithm. *Journal of Chemical Physics*, 109:7694–7702, 1998.

[67] E. L. Pollock and J. Glosli. Comments on P^{3}M, FMM, and the Ewald method for loarge periodic Coulombic systems. *Computer Physics Communications*, 95:93–110, 1996.

[68] W. Shinoda, M. Shiga, and M. Mikami. Rapid estimation of elastic constants by molecular dynamics simulation under constant stress. *Physical Review B*, 69:134103, 2004.

[69] J. J. Cerdà, V. Ballenegger, and C. Holm. Particle-particle particle-mesh method for dipolar interactions: on error estimates and efficiency of schemes with analytical differentiation and mesh interlacing. *Journal of Chemical Physics*, 135:184110, 2011.

[70] R. E. Isele-Holder, W. Mitchell, and A. E. Ismail. Development and application of a particle-particle particle-mesh Ewald method for dispersion interactions. *Journal of Chemical Physics*, 137:174107, 2012.

[71] L. Martínez, R. Andrade, E. G. Birgin, and J. M. Martínez. Packmol: A Package for Building Initial Configurations for Molecular Dynamics Simulations. *Journal of Computational Chemistry*, 30:2157–2164, 2009.

[72] H. J. C. Berendsen, J. R. Grigera, and T. P. Straatsma. The Missing Term in Effective Pair Potentials. *Journal of Physical Chemistry*, 91:6269–6271, 1987.

[73] F. Chen and P. E. Smith. Simulated surface tensions of common water models. *Journal of Chemical Physics*, 126:2211001, 2007.

[74] J. Wang and X. C. Zeng. Computer simulation of liquid-vapor interfacial tension: Lennard-jones fluid and water revisited. *J. Theor. Comput. Chem.*, 8:733–763, 2009.

[75] R. Sakamaki, A. K. Sum, T. Narumi, and K. Yasuoka. Molecular dynamics simulations of vapor/liquid coexistence using the nonpolarizable water models. *Journal of Chemical Physics*, 134:124708, 2011.

[76] M. E. Tuckerman, J. Alejandre, R. López-Rendón, A. L. Jochim, and G. J. Martyna. A Liouville-operator derived measure-preserving integrator for molecular dynamics simulations in the isothermal-isobaric ensemble. *Journal of Physics A - Mathematical and General*, 39:5629–5651, 2006.

[77] J. P. Ryckaert, G. Ciccotti, and H. J. C. Berendsen. Numerical integration of the Cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *Journal of Computational Physics*, 23:327–341, 1977.

[78] A. E. Ismail, G. S. Grest, and M. J. Stevens. Capillary waves at the liquid-vapor interface and the surface tension of water. *Journal of Chemical Physics*, 125:014702, 2006.

[79] M. G. Martin and J. I. Siepmann. Transferable Potentials for Phase Equilibria. 1. United-Atom Description of n-Alkanes. *Journal of Physical Chemistry B*, 102:2569–2577, 1998.

[80] W. R. P. Scott, P. H. Hünenberger, I. G. Tironi, A. E. Mark, S. R. Billeter, J. Fennen, A. E. Torda, T. Huber, P. Krüger, and W. F. van Gunsteren. The GROMOS Biomolecular Simulation Program Package. *Journal of Physical Chemistry A*, 103:3596–3607, 1999.

[81] R. E. Isele-Holder, W. Mitchell, J. R. Hammond, A. Kohlmeyer, and A. E. Ismail. Reconsidering Dispersion Potentials: Reduced Cutoffs in Mesh-Based Ewald Solvers Can Be Faster Than Truncation. *Journal of Chemical Theory and Computation*, 9:5412–5420, 2013.

[82] D. Wolff and W. G. Rudd. Tabulated potentials in molecular dynamics simulations. *Computer Physics Communications*, 120:20–32, 1999.

[83] M. Tuckerman, B. J. Berne, and G. J. Martyna. Reversible multiple time scale molecular dynamics. *Journal of Chemical Physics*, 97:1990–2001, 1992.

[84] S. Plimton, R. Pollock, and M. Stevens. Particle-Mesh Ewald and rRESPA for Parallel Molecular Dynamics Simulations. In *Proceedings of the Eighth SIAM Conference on Parallel Processing for Scientific Computing*, pages 8–21, 1997.

[85] S. Plimpton. Fast Parallel Algorithms for Short-Range Molecular Dynamics. *Journal of Computational Physics*, 117:1–19, 1995.

[86] D. Berthelot. Sur le mélange des gaz. *Comptes Rendus Hebdomadaires des Seances de l'Academie des Sciences*, 126:1703–1855, 1898.

[87] H. A. Lorentz. Ueber die Anwendung des Satzes vom Virial in der kinetischen Theorie der Gase. *Annalen der Physik*, 12:127–136, 1881.

[88] R. J. Good and C. J. Hope. New Combining Rule for Intermolecular Distances in Intermolecular Potential Functions. *Journal of Chemical Physics*, 53:540–543, 1970.

[89] C. L. Kong. Combining rules for intermolecular potential parameters. II. Rules for the Lennard-Jones (12-6) potential and the Morse potential. *Journal of Chemical Physics*, 59:2464–2467, 1973.

[90] M. Waldman and A. T. Hagler. New Combining Rules for Rare Gas van der Waals Parameters. *Journal of Computational Chemistry*, 14:1077–1084, 1993.

[91] F. G. Fumi and M. P. Tosi. Ionic sizes and Born repulsive parameters in NaCl-type Alkali Halides. I. Huggins-Meyer and Pauling Forms. *Journal of Physics and Chemistry of Solids*, 25:31–44, 1964.

[92] M. P. Tosi and F. G. Fumi. Ionic sizes and born repulsive parameters in the NaCl-type alkali halides—II: The generalized Huggins-Mayer form. *Journal of Physics and Chemistry of Solids*, 25:45–52, 1964.

[93] D. S. Cerutti, R. E. Duke, T. A. Darden, and T. P. Lybrand. Staggered Mesh Ewald: An extension of the Smooth Particle-Mesh Ewald method adding great versatility. *Journal of Chemical Theory and Computation*, 5:2322, 2009.

[94] K. P. Ananthapadmanabhan, E. D. Goddard, and P. Chandar. A study of the solution, interfacial and wetting properties of silicone surfactants. *Colloids and Surfaces*, 44:281–297, 1990.

[95] R. Wagner, Y. Wu, G. Czichocki, H. v. Berlepsch, B. Weiland, F. Rexin, and L. Perepelittchenko. Silicon-modified surfactants and wetting: I. Synthesis of the single components of Silwet L77 and their spreading performance on a low-energy solid surface. *Applied Organometallic Chemistry*, 13:611–620, 1999.

[96] R. Wagner, Y. Wu, G. Czichocki, H. v. Berlepsch, F. Rexin, and L. Perepelittchenko. Silicon-modified surfactants and wetting: II. Temperature-dependent spreading behaviour of oligoethylene glycol derivatives of heptamethyltrisiloxane. *Applied Organometallic Chemistry*, 13:201–208, 1999.

[97] R. Wagner, Y. Wu, H. v. Berlepsch, and L. Perepelittchenko. Silicon-modified surfactants and wetting: IV. Spreading behaviour of trisiloxane surfactants on energetically different solid surfaces. *Applied Organometallic Chemistry*, 14:177–188, 2000.

[98] P. E. Theodorakis, E. A. Müller, R. V. Craster, and O. K. Matar. Insights into surfactant-assisted superspreading. *Current Opinion in Colloid & Interface Science*, 2014.

[99] L. Chen, E. Bonaccurso, and M. E. R. Shanahan. Inertial to viscoelastic transition in early drop spreading on soft surfaces. *Langmuir*, 29:1893–8, 2013.

[100] T. D. Blake and J. M. Haynes. Kinematics of liquid/liquid displacement. *Journal of Colloid and Interface Science*, 30:421–423, 1969.

[101] X. Wang, L. Chen, E. Bonaccurso, and J. Venzmer. Dynamic Wetting of Hydrophobic Polymers by Aqueous Surfactant and Superspreader Solutions. *Langmuir*, 29:14855–14864, 2013.

[102] J. Radulovic, K. Sefiane, and M. E. R. Shanahan. Dynamics of Trisiloxane Wetting: Effects of Diffusion and Surface Hydrophobicity. *Journal of Physical Chemistry C*, 114:13620–13629, 2010.

[103] T. Stoebe, Z. Lin, R. M. Hill, M. D. Ward, and H. T. Davis. Superspreading of Aqueous Films Containing Trisiloxane Surfactant on Mineral Oil. *Langmuir*, 13:7282–7286, 1997.

[104] A. Kabalnov. Monolayer frustration contributions to surface and interfacial tensions: explanation of surfactant superspreading. *Langmuir*, 16:2595–2603, 2000.

[105] N. V. Churaev, N. E. Esipova, R. M. Hill, V. D. Sobolev, V. M. Starov, and Z. M. Zorin. The Superspreading Effect of Trisiloxane Surfactant Solutions. *Langmuir*, 17:1338–1348, 2001.

[106] A. Nikolov and D. Wasan. Superspreading mechanisms: An overview. *European Physical Journal - Special Topics*, 197:325–341, 2011.

[107] G. Karapetsas, R. V. Craster, and O. K. Matar. On surfactant-enhanced spreading and superspreading of liquid drops on solid surfaces. *Journal of Fluid Mechanics*, 670:5–37, 2011.

[108] E. Ruckenstein. Superspreading: A possible mechanism. *Colloids and Surfaces A - Physiochemical and Engineering Aspects*, 412:36–37, 2012.

[109] C. Maldarelli. On the microhydrodynamics of superspreading. *Journal of Fluid Mechanics*, 670:1–4, 2011.

[110] A. D. Nikolov, D. T. Wasan, A. Chengara, K. Koczo, G. A. Policello, and I. Kolossvary. Superspreading driven by Marangoni flow. *Advances in Colloid and Interface Science*, 96(1-3):325–38, March 2002.

[111] J. Maxfield and I. W. Shepherd. Conformation of poly(ethylene oxide) in the solid state, melt and solution measured by Raman scattering. *Polymer*, 16:505 – 509, 1975.

[112] S. McNamara, J. Koplik, and J. R. Banavar. Simulations of surfactant-enhanced spreading. *Lecture Notes in Computer Science*, 2073:551–559, 2001.

[113] H.-Y. Kim, Y. Qin, and K. A. Fichthorn. Molecular dynamics simulation of nanodroplet spreading enhanced by linear surfactants. *Journal of Chemical Physics*, 125:174708, 2006.

[114] Y. Shen, A. Couzis, J. Koplik, C. Maldarelli, and M. S. Tomassone. Molecular dynamics study of the influence of surfactant structure on surfactant-facilitated spreading of droplets on solid surfaces. *Langmuir*, 21:12160–70, 2005.

[115] J. D. Halverson, C. Maldarelli, A. Couzis, and J. Koplik. Wetting of hydrophobic substrates by nanodroplets of aqueous trisiloxane and alkyl polyethoxylate surfactant solutions. *Chemical Engineering Science*, 64:4657–4667, 2009.

[116] J. D. Halverson. *Molecular Dynamics Simulations of the Wetting Behavior of Nanodroplets of Water and Aqueous Surfactant Solutions on Solid Substrates of Varying Surface Energy.* Phd thesis, City University of New York, 2008.

[117] O. Borodin and G. D. Smith. Development of Quantum Chemistry-Based Force Fields for Poly(ethylene oxide) with Many-Body Polarization Interactions. *Journal of Physical Chemistry B*, 107:6801–6812, 2003.

[118] J. S. Smith, O. Borodin, and G. D. Smith. A quantum chemistry based force field for poly(dimethoxysiloxane). *Journal of Physical Chemistry B*, 108:20340–20350, 2004.

[119] J. L. F. Abascal and C. Vega. A general purpose model for the condensed phases of water: TIP4P/2005. *Journal of Chemical Physics*, 123:234505, 2005.

[120] C. Vega and J. L. F. Abascal. Simulating water with rigid non-polarizable models: a general perspective. *Physical Chemistry Chemical Physics*, 13:19663–19688, 2011.

[121] J. Fischer, D. Paschek, A. Geiger, and G. Sadowski. Modeling of aqueous poly(oxyethylene) solutions: 1. Atomistic simulations. *Journal of Physical Chemistry B*, 112:2388–98, 2008.

[122] A. Warshel, M. Kato, and A. V. Pisliakov. Polarizable Force Fields: History, Test Cases, and Prospects. *Journal of Chemical Theory and Computation*, 3:2034–2045, 2007.

[123] P. E. M. Lopes, B. Roux, and A. D. Mackerell. Molecular modeling and dynamics studies with explicit inclusion of electronic polarizability. Theory and applications. *Theoretical Chemistry Accounts*, 124:11–28, 2009.

[124] K. A. Motakabbir and M. L. Berkowitz. Liquid–vapor interface of TIP4P water: comparison between a polarizable and a nonpolarizable model. *Chemical Physics Letters*, 176:61–66, 1991.

[125] A. Kohlmeyer, W. Witschel, and E. Spohr. Molecular dynamics simulations of water/metal and water/vacuum interfaces with a polarizable water model. *Chemical Physics*, 213:211–216, 1996.

[126] P. B. Petersen and R. J. Saykally. On the nature of ions at the liquid water surface. *Annual Review of Physical Chemistry*, 57:333–64, 2006.

[127] W. Shinoda, R. DeVane, and M. L. Klein. Multi-property fitting and parameterization of a coarse grained model for aqueous surfactants. *Molecular Simulation*, 33:27–36, 2007.

[128] W. Shinoda, R. DeVane, and M. L. Klein. Coarse-grained molecular modeling of non-ionic surfactant self-assembly. *Soft Matter*, 4:2454, 2008.

[129] M. L. Klein and W. Shinoda. Large-scale molecular dynamics simulations of self-assembling systems. *Science*, 321:798–800, 2008.

[130] W. Shinoda, R. DeVane, and M. L. Klein. Zwitterionic lipid assemblies: molecular dynamics studies of monolayers, bilayers, and vesicles using a new coarse grain force field. *Journal of Physical Chemistry B*, 114:6836–49, 2010.

[131] W. Shinoda, R. DeVane, and M. L. Klein. Coarse-grained force field for ionic surfactants. *Soft Matter*, 7:6178, 2011.

[132] W. G. Noid. Perspective: Coarse-grained models for biomolecular systems. *Journal of Chemical Physics*, 139:090901, 2013.

[133] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein. Comparison of Simple Potential Functions for Simulation Liquid Water. *Journal of Chemical Physics*, 79:926–935, 1983.

[134] C. Oostenbrink, A. Villa, A. E. Mark, and W. F. van Gunsteren. A biomolecular force field based on the free enthalpy of hydration and solvation: the GROMOS

force-field parameter sets 53A5 and 53A6. *Journal of Computational Chemistry*, 25:1656–76, 2004.

[135] K. Vanommeslaeghe, E. Hatcher, C. Acharya, S. Kundu, S. Zhong, J. Shim, E. Darian, O. Guvench, P. Lopes, I. Vorobyov, and A. D. Mackerell. Charmm general force field: A force field for drug-like molecules compatible with the charmm all-atom additive biological force fields. *Journal of Computational Chemistry*, 31:671–690, 2010.

[136] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, and D. A. Case. Development and testing of a general amber force field. *Journal of Computational Chemistry*, 25:1157–74, 2004.

[137] H. Sun and D. Rigby. Polysiloxanes: ab initio force field and structural, conformational and thermophysical properties. *Spectrochimica Acta Part A-Molecular Spectroscopy*, 8539:1301–1323, 1997.

[138] A. L. Frischknecht and J. G. Curro. Improved United Atom Force Field for Poly(dimethylsiloxane). *Macromolecules*, 36:2122–2129, 2003.

[139] E. A. Mason and W. E. Rice. The Intermolecular Potentials of Helium and Hydrogen. *Journal of Chemical Physics*, 22:522, 1954.

[140] G. D. Smith, O. Borodin, and D. Bedrov. A revised quantum chemistry-based potential for poly(ethylene oxide) and its oligomers in aqueous solution. *Journal of Computational Chemistry*, 23:1480–1488, 2002.

[141] D. Bedrov, M. Pekny, and G. D. Smith. Quantum-Chemsitry-Based Force Field for 1,2-Dimethoxyethane and Poly(ethylene oxide) in Aqueous Solution. *Journal of Physical Chemistry B*, 102:996–1001, 1998.

[142] D. Bedrov, O. Borodin, G. D. Smith, F. Trouw, and C. Mayne. Simulation and QENS Studies of Molecular Dynamics in Aqueous Solutions of 1,2-Dimethoxyethane. *Journal of Physical Chemistry B*, 104:5151–5154, 2000.

[143] D. Bedrov and G. D. Smith. Molecular Dynamics Simulations of 1,2-Dimethoxyethane in Aqueous Solution: Influence of the Water Potential. *Journal of Physical Chemistry B*, 103:3791–3796, 1999.

[144] A. E. Ismail, G. S. Grest, D. R. Heine, M. J. Stevens, and M. Tsige. Interfacial Structure and Dynamics of Siloxane Systems: PDMS–Vapor and PDMS–Water. *Macromolecules*, 42:3186–3194, 2009.

[145] R. E. Isele-Holder and A. E. Ismail. Atomistic Potentials for Trisiloxane, Alkyl Ethoxylate, and Perfluoroalkane-Based Surfactants with TIP4P/2005 and Application to Simulations at the Air-Water Interface. *Journal of Physical Chemistry B*, 118:9284–9297, 2014.

[146] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G. A. Petersson, H. Nakatsuji, M. Caricato, X. Li, H. P. Hratchian, A. F. Izmaylov, J. Bloino, G. Zheng, J. L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. A. Montgomery, Jr., J. E. Peralta, F.Ogliaro, M. Bearpark, J. J. Heyd, E. Brothers, K. N. Kudin, V. N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, N. Rega, J. M. Millam, M. Klene, J. E. Knox, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, R. L. Martin, K. Morokuma, V. G. Zakrzewski, G. A. Voth, P. Salvador, J. J. Dannenberg, S. Dapprich, A. D. Daniels, Ö. Farkas, J. B. Foresman, J. V. Ortiz, J. Cioslowski, and D. J. Fox. Gaussian 09 Revision D.01. Gaussian Inc. Wallingford CT 2009.

[147] C. M. Breneman and K. B. Wiberg. Determining atom-centered monopoles from molecular electrostatic potentials. the need for high sampling density in formamide conformational analysis. *Journal of Computational Chemistry*, 11:361–373, 1990.

[148] G. J. Martyna, D. J. Tobias, and M. L. Klein. Constant pressure molecular dynamics algorithms. *Journal of Chemical Physics*, 101:4177–4189, 1994.

[149] M. Parrinello and A. Rahman. Polymorphic transitions in single crystals: A new molecular dynamics method. *Journal of Applioed Physics*, 52:7182–7190, 1981.

[150] E. Alonso, H. Guerrero, D. Montaño, C. Lafuente, and H. Artigas. Thermophysical study of the n-hexane or n-heptane with 1-chloropropane systems. *Thermochimica Acta*, 525:71–77, 2011.

[151] M. Domínguez-Pérez, E. Rilo, L. Segade, C. Franjo, and O. Cabeza. Surface Tension Deviations and Excess Molar Volumes on the Ternary System Propyl Propanoate + Hexane + p-Xylene at 298.15 K. *Journal of Chemical & Engineering Data*, 55:1317–1321, 2010.

[152] F. Veselý, L. Šváb, R. Provazník, and V. Svoboda. Enthalpies of vaporization at high pressures for methanol, ethanol, propan-1-ol, propan-2-ol, hexane, and cyclohexane. *Journal of Chemical Thermodynamics*, 20:981 – 983, 1988.

[153] Y. G. Dobrjakov, I. M. Balashova, and G. Maurer. Experimental Results for the Limiting Activity Coefficients in Some Binary and Ternary Mixtures of Organic Components. *Journal of Chemical & Engineering Data*, 45:185–193, 2000.

[154] M. G. Freire, P. J. Carvalho, A. J. Queimada, I. M. Marrucho, and J. A. P. Coutinho. Surface Tension of Liquid Fluorocompounds. *Journal of Chemical & Engineering Data*, 51:1820–1824, 2006.

[155] E. K. Watkins and W. L. Jorgensen. Perfluoroalkanes: Conformational Analysis and Liquid-State Properties from ab Initio and Monte Carlo Calculations. *Journal of Physical Chemistry A*, 105:4118–4125, 2001.

[156] O. Borodin, R. Douglas, G. D. Smith, F. Trouw, and S. Petrucci. MD Simulations and Experimental Study of Structure, Dynamcs, and Thermodynamics of Poly(ethylene oxide) and Its Oligomers. *Journal of Physical Chemistry B*, 107:6813–6823, 2003.

[157] J. Deng, Y. Yang, P. Wang, G. Ouyang, and Z. Huang. Excess Molar Volumes and Surface Tensions of Trimethylbenzene + Ethylene Glycol Ester at 298.15 K and 313.15 K. *Journal of Chemical & Engineering Data*, 51:725–729, 2006.

[158] D. Patterson, S. N. Bhattacharyya, and P. Picker. Thermodynamics of chain-molecule mixtures: heats of mixing of linear methylsiloxanes. *Transactions of the Faraday Society*, 64:648–661, 1968.

[159] B. Edmonds and I. A. McLure. Thermodynamics of n-alkane + dimethylsiloxane mixtures. part 4.- surface tensions. *Transactions of the Faraday Society*, 78:3319–3329, 1982.

[160] D. F. Wilcock. Vapor pressure-viscosity relations in methylpolysiloxanes. *Journal of the American Chemical Society*, 68:691–696, 1946.

[161] W. Song, P. J. Rossky, and M. Maroncelli. Modeling alkane+perfluoroalkane interactions using all-atom potentials: Failure of the usual combining rules. *Journal of Chemical Physics*, 119:9145–9162, 2003.

[162] L. Lepori, E. Matteoli, A. Spanedda, C. Duce, and M. R. Tiné. Volume changes on mixing perfluoroalkanes with alkanes or ethers at 298.15 K. *Fluid Phase Equilibria*, 201:119–134, 2002.

[163] E. Matteoli and L. Lepori. Determination of the excess enthalpy of binary mixtures from the measurements of the heat of solution of the components: application to the perfluorohexane + hexane mixture. *Fluid Phase Equilibria*, 174:115–131, 2000.

[164] P. J. Daivis and D. J. Evans. Comparison of constant pressure and constant volume nonequilibrium simulations of sheared model decane. *Journal of Chemical Physics*, 100:541, 1994.

[165] B. Das, M. N. Roy, and D. K. Hazra. Densities and viscosities of the binary aqueous mixtures of tetrahydrofuran and 1,2-dimethoxyethane at 298, 308 and 318 K. *Indian Journal of Chemical Technology*, 1:93–97, 1994.

[166] D. Frenkel and B. Smit. *Understanding Molceular Simualtion - From Algorithms to Applications.* Elsevier, second edition, 2002.

[167] T. Schneider and E. Stoll. Molecular-dynamics study of a three-dimensional one-component model for distortive phase transitions. *Physical Review B*, 17:1302–1322, 1978.

[168] S. Cabani, P. Gianni, V. Mollica, and L. Lepori. Groupe contributions to the thermodynamic properties of non-ionic organic solutes in dilute aqueous solution. *Journal of Solution Chemistry*, 10:563–595, 1981.

[169] S. Zeppieri, J. Rodríguez, and A. L. L. de Ramos. Interfacial tension of alkane + water systems. *Journal of Chemical & Engineering Data*, 46:1086–1088, 2001.

[170] N. Nishikido, M. Mahler, and P. Mukerjee. Interfacial tensions of perfluorohexane and perfluorodecalin against water. *Langmuir*, 5:227–229, 1989.

[171] M. Schmidt, AVT-TVT, RWTH Aachen University. (unpublished).

[172] P. T. Kiss and A. Baranyai. A systematic development of a polarizable potential of water. *Journal of Chemical Physics*, 138:204507, 2013.

[173] IAPWS Release on Surface Tension of Ordinary Water Substance. http://www.iapws.org, 1994.

[174] L. S. Bonnington, W. Henderson, and J. A. Zabkiewicz. Characterization of synthetic and commercial trisiloxane surfactant materials. *Applied Organometallic Chemistry*, 18:28–38, 2004.

[175] R. Pan, J. Green, and C. Maldarelli. Theory and Experiment on the Measurement of Kinetic Rate Constants for Surfactant Exchange at an Air/Water Interface. *Journal of Colloid and Interface Science*, 205:213 – 230, 1998.

[176] C. Selve and S. Achilefu. Synthesis of Monodisperse Perfluoroalkyl–Oxyethylene Surfactants with Methoxy Capping: Surfactants of High Chemical Inertness. *Journal of Chemical Society, Chemical Communications*, 13:911–912, 1990.

[177] J. Eastoe, A. Paul, A. Rankin, and R. Wat. Fluorinated Nonionic Surfactants Bearing Either CF_3 – or H–CF_2– Terminal Groups: Adsorption at the Surface of Aqueous Solutions. *Langmuir*, 17:7873–7878, 2001.

[178] J. Eastoe, S. E. Rogers, L. J. Martin, A. Paul, F. Guittard, E. Guittard, R. K. Heenan, and J. R. P. Webster. Fluorosurfactants at Structural Extremes: Adsorption and Aggregation. *Langmuir*, 22:2034–2038, 2006.

[179] C. Jalbert, J. T. Koberstein, I. Yilgor, P. Gallagher, and V. Krukonis. Molecular weight dependence and end-group effects on the surface tension of poly(dimethylsiloxane). *Macromolecules*, 26:3069–3074, 1993.

[180] Y. Xia, N. A. D. Burke, and H. D. H. Stöver. End Group Effect on the Thermal Response of Narrow-Disperse Poly(N -isopropylacrylamide) Prepared by Atom Transfer Radical Polymerization. *Macromolecules*, 39:2275–2283, 2006.

[181] F. P. Zamborini and R. M. Crooks. Corrosion Passivation of Gold by n-Alkanethiol Self-Assembled Monolayers: Effect of Chain Length and End Group. *Langmuir*, 14:3279–3286, 1998.

[182] Y. Xue and M. Ratner. End group effect on electrical transport through individual molecules: A microscopic study. *Physical Review B*, 69:085403, 2004.

[183] T. J. Lenk, D. H. T. Lee, and J. T. Koberstein. End Group Effects on Monolayers of Functionally-Terminated Poly(dimethylsiloxanes) at the Air-Water Interface. *Langmuir*, 10:1857–1864, 1994.

[184] J. R. Lu, R. K. Thomas, and J. Penfold. Surfactant layers at the air/water interface: structure and composition. *Advances in Colloid and Interface Science*, 84:143–304, 2000.

[185] N. Kumar, A. Couzis, and C. Maldarelli. Measurement of the kinetic rate constants for the adsorption of superspreading trisiloxanes to an air/aqueous interface and the relevance of these measurements to the mechanism of superspreading. *Journal of Colloid and Interface Science*, 267:272–285, 2003.

[186] D. R. Heine, G. S. Grest, and E. B. Webb, III. Spreading dynamics of polymer nanodroplets in cylindrical geometries. *Physical Review E*, 70:011606, 2004.

[187] N. A. Ivanova, Z. B. Zhantenova, and V. M. Starov. Wetting dynamics of polyoxyethylene alkyl ethers and trisiloxanes in respect of polyoxyethylene chains and properties of substrates. *Colloids and Surfaces A - Physiochemical and Engineering Aspects*, 413:307–313, 2012.

[188] C. J. Van Oss, R. J. Good, and R. J. Busscher. Estimation of the Polar Surface Tension Parameters of Glycerol and Formamide, for Use in Contact Angle Measurements on Polar Solids. *Journal of Dispersion Science and Technology*, 11:75–81, 1990.

[189] MAPS platform. Scienomics. http://www.scienomics.com.

[190] R. E. Isele-Holder, B. Berkels, and A. E. Ismail. Smoothing of contact lines in spreading droplets by trisiloxane surfactants and its relevance for superspreading. *Soft Matter*, 11:4527–4539, 2015.

[191] D. R. Heine, G. S. Grest, and E. B. Webb, III. Spreading dynamics of polymer nanodroplets. *Physical Review E*, 68:061603, 2003.

[192] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42:577–685, 1989.

[193] B. Berkels, I. Cabrilo, S. Haller, M. Rumpf, and K. Schaller. Co-registration of intra-operative brain surface photographs and pre-operative MR images. *International Journal of Computer Assisted Surgery and Radiology*, 9:387–400, 2014.

[194] B. Berkels. An unconstrained multiphase thresholding approach for image segmentation. *Lecture Notes in Computer Science*, 5567:26–37, 2009.

[195] A. Chambolle and T. Pock. A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. *Journal of Mathematical Imaging and Vision*, 40:120–145, 2010.

[196] AG Rumpf, Institute for Numerical Simulation, University of Bonn. Quocmesh software library. http://numod.ins.uni-bonn.de/software/quocmesh/.

[197] H. A. Ritacco, F. Ortega, R. G. Rubio, N. Ivanova, and V. M. Starov. Equilibrium and dynamic surface properties of trisiloxane aqueous solutions. *Colloids and Surfaces A - Physiochemical and Engineering Aspects*, 365:199–203, 2010.

[198] M. G. Schmidt, R. A. Sauer, and A. E. Ismail. Multiscale treatment of mechanical contact problems involving thin polymeric layers. *Modelling and Simulation in Materials Science and Engineering*, 22:045012, 2014.

[199] V. M. Samsonov. On computer simulation of droplet spreading. *Current Opinion in Colloid & Interface Science*, 16:303–309, 2011.

[200] S. Bardon, M. Cachile, A.-M. Cazabat, X. Fanton, M.-P. Valignat, and S. Villette. Structure and dynamics of liquid films on solid surfaces. *Faraday Discussions*, 104:307, 1996.

[201] A.-M. Cazabat, N. Fraysse, F. Heslot, P. Levinson, J. Marsh, F. Tiberg, and M. P. Valignat. Pancakes. *Advances in Colloid and Interface Science*, 48:1–17, April 1994.

[202] U. Albrecht, A. Otto, and P. Leiderer. Two-dimensional liquid polymer diffusion: Experiment and simulation. *Physical Review Letters*, 68:3192–3195, 1992.

[203] P.-G. De Gennes and A.-M. Cazabat. Étalement d'une goutte stratifiée incompressible. *Comptes Rendus de l'Academie des Sciences*, 310:1601–1606, 1990.

[204] D. B. Abraham, P. Collet, J. De Coninck, and F. Dunlop. Langevin dynamics of spreading and wetting. *Physical Review Letters*, 65:195–198, 1990.

[205] D. Abraham, P. Collet, J. Coninck, and F. Dunlop. Langevin Dynamics of an Interface near a Wall. *Journal of Statistical Physics*, 61:509–532, 1990.

[206] S. F. Burlatsky, G. Oshanin, A.-M. Cazabat, and M. Moreau. Microscopic model of upward creep of an ultrathin wetting film. *Physical Review Letters*, 76:86–89, 1996.

[207] S. F. Burlatsky, G. Oshanin, A.-M. Cazabat, M. Moreau, and W. P. Reinhardt. Spreading of a thin wetting film: Microscopic approach. *Physical Review E*, 54:3832–3845, 1996.

[208] M. Popescu and S. Dietrich. Model for spreading of liquid monolayers. *Physical Review E*, 69:061602, 2004.

[209] A. Lukkarinen, K. Kaski, and D. B. Abraham. Mechanisms of fluid spreading: Ising model simulations. *Physical Review E*, 51:2199–2202, 1995.

[210] D. B. Abraham, R. Cuerno, and E. Moro. Microscopic Model for thin film spreading. *Physical Review Letters*, 88:206101, 2002.

[211] J.-X. Yang, J. Koplik, and J. Banavar. Molecular dynamics of drop spreading on a solid surface. *Physical Review Letters*, 67:3539–3542, 1991.

[212] J.-X. Yang, J. Koplik, and J. Banavar. Terraced spreading of simple liquids on solid surfaces. *Physical Review A*, 46:7738–7749, 1992.

[213] S. Bekink, S. Karaborni, G. Verbist, and K. Esselink. Simulating the Spreading of a Drop in the Terraced Wetting Regime. *Physical Review Letters*, 76:3766–3769, 1996.

[214] J. A. Nieminen, D. B. Abraham, M. Karttunen, and K. Kaski. Molecular dynamics of a microscopic droplet on solid surface. *Physical Review Letters*, 69:124–127, 1992.

[215] J. A. Nieminen and T. Ala-Nissila. Spreading dynamics of polymer microdroplets: A molecular-dynamics study. *Physical Review E*, 49:4228–4236, 1994.

[216] M. Voué, M. P. Valignat, G. Oshanin, A.-M. Cazabat, and J. De Coninck. Dynamics of spreading of liquid microdroplets on substrates of increasing surface energies. *Langmuir*, 14:5951–5958, 1998.

[217] E. A. Mastny and J. J. de Pablo. Melting line of the Lennard-Jones system, infinite size, and full potential. *Journal of Chemical Physics*, 127:104504, 2007.

[218] J. De Coninck, U. D'Ortona, J. Koplik, and J. R. Banavar. Terraced Spreading of Chain Molecules via Molecular Dynamics. *Physical Review Letters*, 74:928–931, 1995.

[219] U. D'Ortona, J. De Coninck, J. Koplik, and J. Banavar. Terraced spreading mechanisms for chain molecules. *Physical Review E*, 53:562–569, 1996.

[220] D. R. Heine, G. S. Grest, and E. B. Webb, III. Surface Wetting of Liquid Nanodroplets: Droplet-Size Effects. *Physical Review Letters*, 95:107801, 2005.

[221] J. D. Halverson, W. B. Lee, G. S. Grest, A. Y. Grosberg, and K. Kremer. Molecular dynamics simulation study of nonconcatenated ring polymers in a melt. II. Dynamics. *Journal of Chemical Physics*, 134:204905, 2011.

[222] Co. Wu, T. Qian, and P. Sheng. Droplet spreading driven by van der Waals force: a molecular dynamics study. *Journal of Physics: Condensed Matter*, 22:325101, 2010.

[223] M. Voué and J. De Coninck. Spreading and wetting at the microscopic scale: recent developments and perspectives. *Acta Materialia*, 48:4405 – 4417, 2000.

[224] M. Voué, S. Rovillard, J. De Coninck, M. P. Valignat, and A.-M. Cazabat. Spreading of Liquid Mixtures at the Microscopic Scale: A Molecular Dynamics Study of the Surface-Induced Segregation Process. *Langmuir*, 16:1428–1435, 2000.

[225] H. Xu, D. Shirvanyants, K. Beers, K. Matyjaszewski, M. Rubinstein, and S. Sheiko. Molecular Motion in a Spreading Precursor Film. *Physical Review Letters*, 93:206103, 2004.

[226] H. Xu, F. C. Sun, D. G. Shirvanyants, M. Rubinstein, D. Shabratov, K. L. Beers, K. Matyjaszewski, and S. S. Sheiko. Molecular Pressure Sensors. *Advanced Materials*, 19:2930–2934, 2007.

[227] K. Kremer and G. S. Grest. Dynamics of entangled linear polymer melts: A molecular-dynamics simulation. *Journal of Chemical Physics*, 92:5057, 1990.

[228] A. Lotfi, J. Vrabec, and J. Fischer. Vapour liquid equilibria of the Lennard-Jones fluid from the NpT plus test particle method. *Molecular Physics*, 76:1319–1333, 1992.

[229] O. Borodin and G. D. Smith. Development of many-body polarizable force fields for Li-battery components: 1. Ether, alkane, and carbonate-based solvents. *Journal of Physical Chemistry B*, 110:6279–92, 2006.

[230] F. Leroy. Molecular dynamics calculations of solid-liquid interfacial tensions. 27th European Symposium on Applied Thermodynamics (ESAT 2014): “Experiments meet Theory and Simulation” 6-9 July 2014, Eindhoven - The Netherlands, 2014.

[231] S.-T. Lin, M. Blanco, and W. A. Goddard, III. The two-phase model for calculating thermodynamic properties of liquids from molecular dynamics: Validation

for the phase diagram of Lennard-Jones fluids. *Journal of Chemical Physics*, 119:11792, 2003.

[232] S.-T. Lin, P. K. Maiti, and W. A. Goddard, III. Two-phase thermodynamic model for efficient and accurate absolute entropy of water from molecular dynamics simulations. *Journal of Physical Chemistry B*, 114:8191–8, 2010.

[233] T. A. Pascal, S.-T. Lin, and W. A. Goddard, III. Thermodynamics of liquids: standard molar entropies and heat capacities of common solvents from 2PT molecular dynamics. *Physical Chemistry Chemical Physics*, 13:169–81, 2011.

[234] T.-S. Lin, T. A. Pascal, M. Blanco, and W. A. Goddard, III. Two Phase Thermodynamics (2PT) Program for entropy and Free Energies of Complex Systems. (unpublished).

[235] N. G. Hadjiconstantinou. Discussion of recent developments in hybrid atomistic-continuum methods for multiscale hydrodynamics. *Bulletin of the Polish Academy of Sciences Technical Sciences*, 53:335–342, 2005.

[236] K. M. Mohamed and A. A. Mohamad. A review of the development of hybrid atomistic-continuum methods for dense fluids. *Microfluidics and Nanofluidics*, 8:283–302, 2010.

[237] H. S. Wijesinghe and N. G. Hadjiconstantinou. Discussion of Hybrid Atomistic-Continuum Methods for Multiscale Hydrodynamics. *International Journal for Multiscale Computational Engineering*, 2:189–202, 2004.

[238] R. C. Tolman. Consideration of the Gibbs Theory of Surface Tension. *Journal of Chemical Physics*, 16:758–774, 1948.

[239] J. G. Kirkwood and F. P. Buff. The Statistical Mechanical Theory of Surface Tension. *Journal of Chemical Physics*, 17:338–343, 1949.

[240] J. S. Huang and W. W. Webb. Diffuse Interface in a Critical Fluid Mixture. *Journal of Chemical Physics*, 50:3677–3693, 1969.

[241] D. Beysens and M. Robert. Thickness of fluid interfaces near the critical-point from optical reflecitvity measurements. *Journal of Chemical Physics*, 87:3056–3061, 1987.

[242] S. W. Sides, G. S. Grest, and M.-D. Lacasse. Capillary waves at liquid-vapor interfaces: A molecular dynamics simulation. *Physical Review E*, 60:6708–6713, 1999.

[243] H. Flyvbjerg and H. G. Petersen. Error estimates on averages of correlated data. *Journal of Chemical Physics*, 91(July):461–466, 1989.

[244] G. D. Smith, R. L. Jaffe, and D. Y. Yoon. Force field for simulations of 1,2-dimethoxyethane and poly(oxyethylene) based upon ab initio electronic structure calculations on model molecules. *Journal of Physical Chemistry*, 97:12752–12759, 1993.