

Bachelor's Thesis

Teilchensimulationen von Polymermischungen in begrenzten Geometrien mit zeitabhängigen Randbedingungen

Particle simulations of polymer mixtures in confined geometries with time dependent boundary conditions

prepared by
Justus Multhaup

from Holzminden

at the Institut für Theoretische Physik

Thesis period: 8th May 2023 until 14th August 2023

First Referee: Prof. Dr. Marcus Müller

Second referee: Prof. Dr. Stefan Klumpp

Contents

1	Introduction	1
2	Theory	4
2.1	Single-chain properties	4
2.1.1	Polymer types	4
2.1.2	The freely jointed chain	4
2.1.3	Rouse model	5
2.2	Polymeric mixtures	6
2.2.1	Flory-Huggins Theory	7
2.2.2	Phase separation	8
2.3	Collective diffusion	8
2.4	Single chain in mean field simulations	11
2.5	Smart Monte Carlo Move	13
2.6	Discrete optimization	14
2.7	Simulated Annealing	14
3	Simulation technique	16
4	Collective diffusion of symmetric homopolymers	18
4.1	Reference system	18
4.2	Collective diffusion coefficient	20
5	External-field-driven flow	24
5.1	Reference system	25
5.2	Bending modulus	26
6	Polymer-type-conversion-based target composition	31
6.1	Formulation of the problem	31
6.2	Steepest Descent	37
6.3	Simulated annealing	37

Contents

6.4	Comparison of algorithms	38
6.4.1	SA parameter choice	39
6.5	SD simulations	43
6.5.1	Impact on bulk	45
6.6	Hybrid strategy	45
7	Summary	47
7.1	Collective diffusion	47
7.2	External field driven flow	47
7.3	Polymer-type-conversion-based target composition	48
8	Outlook	49

Nomenclature

Acronyms

EC	Exponential Cooling	39
GPU	Graphics processing unit	16
LC	Linear Cooling	39
MC	Monte-Carlo	1
MD	Molecular Dynamics	1
MPI	Message Passing Interface	16
NESS	Non-equilibrium steady state	18
ODT	Order-to-disorder transition	8
OOT	Order-to-order transition	8
SA	Simulated Annealing	3
SCFT	Self-Consistent Field Theory	8
SCMF	Single-Chain-in-Mean-Field	2
SD	Steepest Descent	3
SMC	Smart Monte-Carlo	13
SOMA	SOft coarse-grained Monte-Carlo Acceleration	2
SSL	Strong-segregation limit	8
WSL	Weak-segregation limit	8

1 Introduction

Polymers are highly versatile macromolecules composed of repeating units called monomers. Their importance for life on earth cannot be overstated, biological polymers are involved in countless chemical reactions in the human body and form the backbone of proteins and DNA. Synthetic polymers are applied almost everywhere in modern society, ranging from simple packaging materials to highly sophisticated materials used in aerospace engineering. This is owed to their vast range of unique physical properties, such as high elasticity, flexibility and durability. When creating novel materials with very precise features, understanding how polymers interact with one another and form structure via phase separation in multicomponent systems is essential.

One important class of polymers is the so-called homopolymer, which is made of a single type of monomer unit, for example, polyethylene. Copolymers, in contrast, consist of two or more different types of monomer units. Copolymers that are concatenations of homopolymers of different types and lengths are called block-copolymers. One of the most interesting and extensively studied properties of copolymers is their ability to self-assemble into microphases [1]. An important tool in understanding the mechanisms behind this microphase separation, and many other interesting properties, is the use of computer simulations. These include particle-based Molecular Dynamics (MD) and Monte-Carlo (MC) methods as well as continuum-model-based and hybrid methods [2]. Particle-based simulations accurately model small-scale phenomena. However, despite rapid advancements in computer technology and high parallelizability, they are too computationally expensive to simulate long-timescale phenomena since they involve the explicit calculation of the strong, bonded interactions between particles. This also poses limitations on the system size; macro-scale phenomena are hard to capture. Continuum models, on the other hand, are less computationally expensive, but at the cost of a lower accuracy on the microscale. On the mesoscale, good agreement between the two types of models was observed, e.g. in the orientation of cylindrical mesophases upon sol-

1 Introduction

vent evaporation [3]. To evaluate the accuracy of continuum models on small-length scales, it is necessary to compare them to established particle-based models. One approach to this is to consider a simulation box in a particle-based simulation as a small section from a large continuum-model-based simulation. The dynamics are then driven by extracting the time evolution of the density fields at the boundaries from the continuum simulation and applying them in the particle-based simulation, e.g. employing non-periodic boundary conditions. However, dictating the boundary densities in particle-based simulations is not a trivial task. One option is the use of external fields or the umbrella sampling method [4], although this has the large drawback that the number of particles in the simulation box remains constant, while in the continuum simulation particles can enter and leave the section at any time. An alternative is to employ conversion zones at the boundaries, in which molecules are converted to different types, mimicking particle exchange with the surrounding as shown in Figure 1.1. If the densities are defined on a discretized grid, this leads to an optimization problem in which the ideal configuration of molecule types needs to be found, such that the mean squared deviation from the target composition is minimized.

This thesis is divided into three parts. All simulations are carried out using the

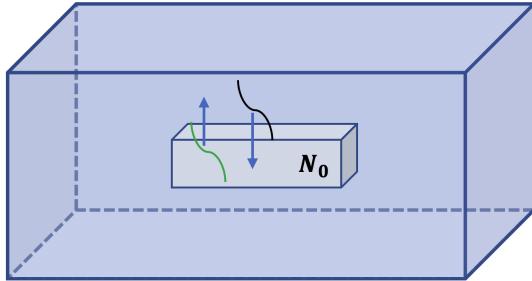


Figure 1.1: Sketch of the setting explained in the text. Polymers of different types may be exchanged with the surroundings, but the number of molecules N_0 inside the section of the particle simulation remains constant.

SOft coarse-grained Monte-Carlo Acceleration (SOMA) [5] software package based on the Single-Chain-in-Mean-Field (SCMF) algorithm [6]. In the first part, as a first step towards boundary-driven particle simulations, the dynamics of noninteracting homopolymers are investigated. Specifically, a system is pushed away from equilibrium by introducing conversion zones at the boundary regions of the simulation box. This results in a diffusive current and allows the calculation of the collective diffusion coefficient of the system. To this end, the .

1 Introduction

In the second part, a method to implement time-dependent boundary conditions in SOMA is introduced using piecewise external fields. This method is exploited to obtain the bending modulus of diblock-copolymers.

In the final part, two approaches to dictate the composition at the boundaries according to a target field are introduced. The first approach is based on the conversion of macromolecule types and results in a non-convex optimization problem, as mentioned before. Two optimization algorithms are presented: the Simulated Annealing (SA) algorithm and the Steepest Descent (SD) algorithm. In the second approach, the applied MC scheme is modified in addition to the macromolecule type conversions to give a hybrid scheme.

2 Theory

2.1 Single-chain properties

Since the collective dynamics of polymer mixtures can often be directly related to the single-chain properties, some selected aspects are briefly discussed here. The simplest model to describe a polymer chain is the ideal chain, for which any non-bonded interaction between monomers is neglected.

2.1.1 Polymer types

The properties of polymers depend greatly on their molecular architecture. The simplest type of polymer is the homopolymer, which is a linear chain of monomers of the same monomer type.

- Show picture of different polymer types here

2.1.2 The freely jointed chain

The freely jointed chain model is an ideal model in which the chain is composed of N segments of length b_0 , whose orientation vectors \mathbf{r}_i ($b_0 = |\mathbf{r}_i|$) can face in any direction and are completely independent from each other. Each segment therefore represents a step in a random walk [7]. The chain configuration may be characterised by the end-to-end vector \mathbf{R} :

$$\mathbf{R} = \sum_{i=1}^N \mathbf{r}_i , \quad (2.1)$$

connecting the first and last segments. This is shown in Figure 2.1. Due to the completely random orientation, \mathbf{R} has zero mean. A useful property to quantify the spatial extension of the chain is the mean square of \mathbf{R} , which calculates to:

$$R_e^2 \equiv \langle \mathbf{R}^2 \rangle = Nb_0^2. \quad (2.2)$$

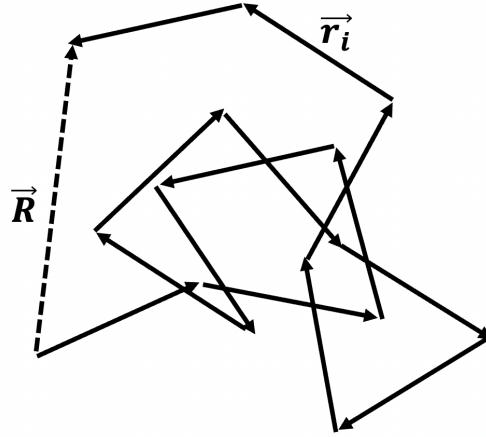


Figure 2.1: Freely jointed chain.

For large N , \mathbf{R} has a Gaussian distribution [8]:

$$P(N, \mathbf{R}) = \left(\frac{3}{2\pi R_e^2} \right)^{3/2} \exp \left[-\frac{3\mathbf{R}^2}{2R_e^2} \right], \quad (2.3)$$

the freely jointed chain therefore converges towards the Gaussian chain.

2.1.3 Rouse model

The Rouse model [9] is used to describe the dynamics of a Gaussian chain moving through a solvent. The stiff bonds are replaced by springs of root-mean-square size b_0 , through which a monomer interacts with its neighbours. The monomers experience a friction force with friction coefficient ζ , and the total friction coefficient of the center of mass is:

$$\zeta_R = N\zeta. \quad (2.4)$$

Within the Rouse model, the diffusion coefficient of the chain is computed as:

2 Theory

$$D_R = \frac{k_B T}{\zeta_R} = \frac{k_B T}{N \zeta}. \quad (2.5)$$

For an n -dimensional diffusion of a Brownian particle, the single chain diffusion constant is related to the mean squared displacement of the chain center of mass $g_3(t)$ by the Einstein relation [10]:

$$\begin{aligned} D &= \frac{g_3(t)}{2nt}, \\ g_3(t) &= \langle (\mathbf{r}_{cm}(t) - \mathbf{r}_{cm}(t_0))^2 \rangle, \end{aligned} \quad (2.6)$$

where $\mathbf{r}_{cm}(t)$ is the center of mass position at time t and t_0 is the time at which the measurement is started. The time in which a polymer diffuses a distance of the order of its size is called relaxation time τ_R :

$$\tau_R \approx \frac{R_e^2}{D_R} = \frac{\zeta}{k_B T} N R_e^2. \quad (2.7)$$

On time scales larger than τ_R , the chain movement is only due to diffusion, while viscoelastic effects are observed at shorter time scales.

2.2 Polymeric mixtures

Polymer mixtures consist of two or more chemically different polymer types. The mechanical and thermodynamic properties can vary greatly with several factors such as density, molecular weight and interactions between the polymers. This makes them desirable for manufacturing materials with tailored properties.

If the density is uniform everywhere, the mixture is called homogeneous. In a heterogeneous mixture, in contrast, the density is non-uniform, leading to visible boundaries which may have very different properties. This phenomenon is also called macro-phase separation. From an entropic viewpoint, mixing is always favored. However, energetic interactions between polymers may either favor or suppress mixing. Whether a mixture is homogeneous or heterogeneous, thus, depends on the balance between entropy and energy [8, S. 137].

2.2.1 Flory-Huggins Theory

Whether mixing or phase separation will be favored can be predicted by determining the free-energy change associated with mixing the components. This free energy-change can be computed within the lattice model developed by Flory and Huggins [11]. Within the Flory-Huggins framework, no volume change is assumed upon mixing. With this assumption, it is convenient to represent the system on a lattice. A lattice site corresponds to the smallest molecular unit and every macromolecule takes up one or multiple lattice sites. Consider a binary mixture with n_A polymers of species A and chain length N_A and n_B polymers of species B and chain length N_B . Let the total number of polymers be $n = n_A + n_B$. The free energy of mixing per monomer ΔF_{mix} is then given by the Flory-Huggins equation of polymer solutions [8, S. 143]:

$$\frac{\Delta F_{mix}}{k_B T} = \frac{\phi}{N_A} \ln \phi + \frac{1 - \phi}{N_B} \ln(1 - \phi) + \chi \phi(1 - \phi). \quad (2.8)$$

Here, $\phi = n_A N_A / (n_A N_A + n_B N_B)$ is the monomer fraction of species A, k_B is the Boltzmann constant, T is the system temperature and χ is the Flory interaction parameter which characterizes the interaction between different polymer species and can be obtained from experiments. A positive value of χ opposes mixing while a negative value promotes it; knowing the value of χ , therefore, allows us to qualitatively predict the phase separation behavior. For $\chi = 0$, Equation 2.8 reduces to regular solution theory for an ideal gas of polymers with concentrations ϕ/N_A and $(1 - \phi)/N_B$. Note that so far, no space dependency of ϕ has been assumed. In the following discussion, a symmetric mixture with $N_A = N_B = N$ is assumed. To fully capture the complexity of the system, the Flory-Huggins model has to be extended to include spatial variations of ϕ , which gives rise to the de Gennes-Flory-Huggins free-energy functional [12, 13]:

$$\frac{F[\phi] R_e^3}{k_B T \sqrt{\bar{N}}} = \int d^3 r \left\{ \phi \ln \phi + (1 - \phi) \ln(1 - \phi) + \chi N \phi(1 - \phi) + k(\phi)[\nabla \phi]^2 \right\}. \quad (2.9)$$

Here, $\bar{N} = (n R_e^3 / V)^2$ is the invariant degree of polymerization of the system with volume V . It is a measure of the number of neighboring chains a chain interacts with. The term proportional to $[\nabla \phi]^2$ is added to the free-energy density to ensure that

2 Theory

unphysical, sharp changes in the local densities are penalized. The precise form of $k(\phi)$ depends on the strength of the parameter χ . For small $\chi N \lesssim 5$, one considers the Weak-segregation limit (WSL). For large $\chi N \gtrsim 10$ [14], the Strong-segregation limit (SSL) holds. In these two limits, the prefactor k takes the following form [13]:

$$k_{\text{WSL}} = \frac{R_e^2}{36\phi(1-\phi)}; \quad k_{\text{SSL}} = \frac{R_e^2}{24\phi(1-\phi)}. \quad (2.10)$$

2.2.2 Phase separation

The incompatibility between different monomer types causes diblock copolymers to self-assemble into interesting equilibrium morphologies. Besides the Flory-Huggins parameter χN , the volume fraction f of the A block is crucial in determining the phase-separation behavior. Typical morphologies are shown in Figure 2.2 for increasing f . The corresponding phase diagram may be obtained from experiments, or, numerically, from Self-Consistent Field Theory (SCFT) computations [15]. The result of the latter is shown in Figure 2.2.

Below a critical value of $\chi N \approx 10.5$, no structure formation occurs. This value is called the Order-to-disorder transition (ODT). At a fixed $\chi N > 10.5$, several transitions between distinct, ordered morphologies occur for increasing f . These transitions are called Order-to-order transitions (OOTs). The A blocks first form closely packed spheres (CPS), which then transition into body-centered cubic spheres (S) and for even larger f into cylindrical phases (C). Before reaching the lamellar phase (L), for not too large values of χN , an additional phase is observed, the bicontinuous gyroids (G). The same transitions occur in reversed order if f is increased further beyond 0.5, where the characteristic morphologies are now assumed by the B blocks.

2.3 Collective diffusion

Consider again a binary mixture of homopolymers with $N_A = N_B = N$. Since the number of polymers in the system is constant, the continuity equation holds:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{J}_A = 0. \quad (2.11)$$

2 Theory

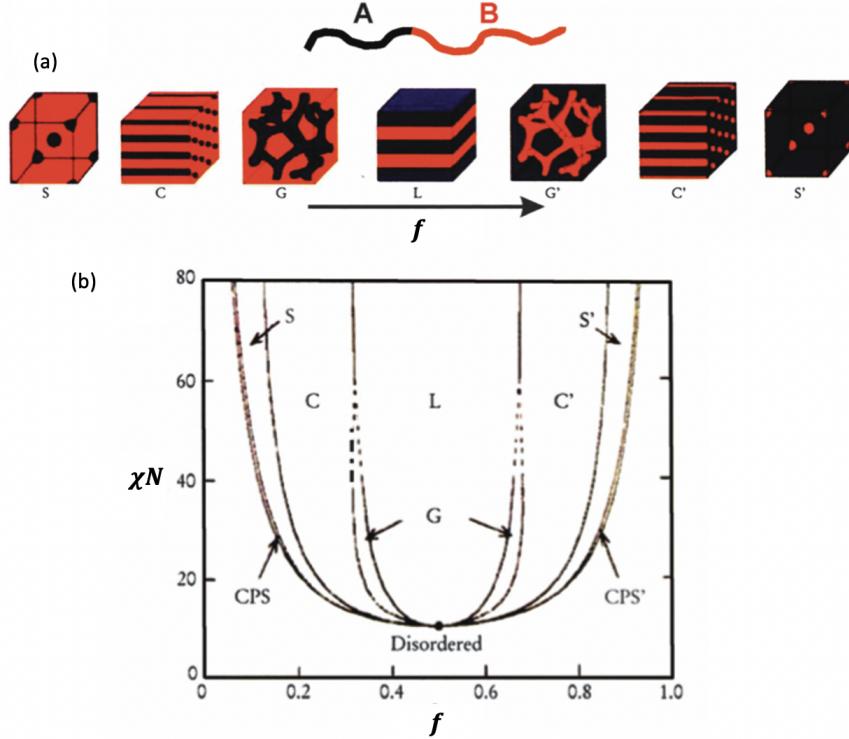


Figure 2.2: (a) Different equilibrium morphologies of diblock-copolymers for increasing values of f . (b) χN - f phase diagram. CPS stands for closely packed spheres, S for body-centered cubic spheres, C for hexagonally packed cylinders, G for bicontinuous gyroids and L for lamelae. Adapted from [16].

Here, \mathbf{J}_A is the local current of polymer species A. Near equilibrium, one postulates a linear relation between \mathbf{J}_A and the local chemical potential gradient $\nabla\mu_A$ [12]. In the most general case, this relation is non-local in space and time. Assuming translational invariance in space and time, it reads [17]:

$$\mathbf{J}_A(\mathbf{r}, t) = - \int_{t'=-\infty}^t dt' \int_V d\mathbf{r}' \frac{\tilde{\Lambda}_A(\mathbf{r} - \mathbf{r}', t - t')}{k_B T} \nabla' \mu_A(\mathbf{r}', t'). \quad (2.12)$$

The Onsager coefficient $\tilde{\Lambda}_A(\mathbf{r} - \mathbf{r}', t - t')$ relates the gradient of the chemical potential at position \mathbf{r}' to a density flux at position \mathbf{r} and also accounts for memory effects. It is often a good approximation to neglect the transmission of forces along the molecular backbone of the chain [18]. In this local approximation, the Onsager coefficient takes the simple form $\tilde{\Lambda}_A(\mathbf{r}, t) = \Lambda_A(\mathbf{r}, t)\delta(\mathbf{r} - \mathbf{r}')\delta(t - t')$, so

2 Theory

Equation 2.12 simplifies to:

$$\mathbf{J}_A = -\frac{\Lambda_A}{k_B T} \nabla \mu_A . \quad (2.13)$$

This approximation leads to a simple form of the Onsager coefficient for incompressible dynamics that will be derived in the following.

Following the discussion in the appendix of [12], consider a mixture of polymer chains with densities ϕ_A and ϕ_B . Incompressibility, e.g. $\phi_A + \phi_B = 1$, is enforced by introducing an additional, repulsive potential U to the chemical potential, so from Equation 2.13 one obtains:

$$\mathbf{J}_A = -\Lambda_A \nabla [(\mu_A + U)/k_B T] , \quad (2.14a)$$

$$\mathbf{J}_B = -\Lambda_B \nabla [(\mu_B + U)/k_B T] . \quad (2.14b)$$

Due to the incompressibility, the total current $\mathbf{J}_{tot} = \mathbf{J}_A + \mathbf{J}_B$ must have zero divergence, which in one dimension means $\mathbf{J}_{tot} \equiv J_{tot} \mathbf{e}_x = \text{const}$. From the Galilei invariance of the system, we can simply choose $J_{tot} = 0$. From this condition, U can be calculated explicitly:

$$U = (\Lambda_A \mu_A + \Lambda_B \mu_B) / (\Lambda_A + \Lambda_B) . \quad (2.15)$$

Since one of the currents is redundant, write $J \equiv J_A$. From Equation 2.14, obtain:

$$J = -\frac{\Lambda}{k_B T} \nabla (\mu_A - \mu_B) \equiv -\frac{\Lambda}{k_B T} \nabla \mu , \quad (2.16)$$

where $\Lambda = \Lambda_A \Lambda_B / (\Lambda_A + \Lambda_B)$ and μ is the exchange chemical potential. In the limit of no interactions, the mixture can be considered as an ideal gas, so $\Lambda_i = D_i \phi_i$ for $i = A, B$. For $D_A = D_B \equiv D$, this yields:

$$\Lambda = D \phi(\mathbf{r}) (1 - \phi(\mathbf{r})) . \quad (2.17)$$

2 Theory

This approximation is frequently used in the literature for the sake of computational efficiency [12, 19, 20]. It corresponds to a local coupling in which monomers move like the center of mass, therefore it does not account for the connectivity of the monomers along the backbone of the chain.

2.4 Single chain in mean field simulations

Although the model used in this thesis provides a high flexibility in molecular architectures, the following discussion is restricted to symmetric AB-diblock copolymers in a volume V at temperature T for simplicity. We consider the partition function [6]:

$$\mathcal{Z} \propto \frac{1}{n!} \int \prod_{i=1}^n \mathcal{D}[\mathbf{r}_i(s)] \mathcal{P}_0[\mathbf{r}_i(s)] \exp\left(\frac{-\mathcal{H}_{nb}[\hat{\phi}_A, \hat{\phi}_B]}{k_B T}\right), \quad (2.18)$$

where $\mathbf{r}_i(s)$ is the position of segment s from molecule i . The factor $\int \mathcal{D}[\mathbf{r}_i(s)] \equiv \int \prod_s d^3 \mathbf{r}_i(s)$ integrates over the conformation space of molecule i . Conformations are weighted by a Boltzmann factor $P_0[\mathbf{r}_i(s)] \propto \exp(-\mathcal{H}_b[\mathbf{r}_i(s)]/k_B T)$ that involves the bonded interactions in the absence of any intermolecular interactions. The bonded interactions $\mathcal{H}_b[\mathbf{r}_i(s)]$ are modeled by harmonic potentials, which is appropriate for Gaussian polymers. The discretized form reads:

$$\frac{\mathcal{H}_b[\mathbf{r}_i(s)]}{k_B T} = \sum_{s=1}^{N-1} \frac{3(N-1)}{2R_{e0}^2} [\mathbf{r}_i(s) - \mathbf{r}_i(s+1)]. \quad (2.19)$$

Here, R_{e0}^2 is the mean squared end-to-end distance of the polymer in the absence of non-bonded interactions. The non-bonded interactions \mathcal{H}_{nb} in Equation 2.18 depend on the local particle densities $\hat{\phi}_A$ and $\hat{\phi}_B$, which are defined on a cubic grid with spacing ΔL in the discretized model, the total number of cells is $N_{cells} = V/(\Delta L)^3$. The densities are obtained by summing over all beads and assigning them to their according cell:

$$\hat{\phi}_{\alpha,m} = \sum_{i=1}^n \sum_{s=1}^N \Pi(\mathbf{r}_i(s), \mathbf{c}_m) \gamma_\alpha(s), \quad (2.20)$$

2 Theory

where $\gamma_\alpha(s)$ assigns the segments to their respective types: it is one if segment s is of type α , and zero otherwise. $\Pi(\mathbf{r}_i(s), \mathbf{c}_m)$ is the characteristic function of the grid cell centered at \mathbf{c}_m . The non-bonded interactions are modeled by the following discretized excess free energy functional:

$$\frac{\mathcal{H}_{\text{nb}}[\hat{\phi}_A, \hat{\phi}_B]}{k_B T} = \frac{\rho_0 \Delta L^3}{N} \sum_{m=1}^{N_{\text{cells}}} \left(\frac{\kappa_0 N}{2} [\hat{\phi}_{A,m} + \hat{\phi}_{B,m} - 1]^2 - \frac{\chi_0 N}{4} [\hat{\phi}_{A,m} - \hat{\phi}_{B,m}]^2 \right). \quad (2.21)$$

where $\rho_0 = \frac{nN}{V}$ is the average monomer density. The first term in the sum penalizes deviations of the total density from unity, and therefore controls the extent of density fluctuations with the control parameter κ_0 . The second term enforces the repulsion of monomers of different types with the control parameter χ_0 . It should be emphasized that R_{e0}^2 , $\kappa_0 N$ and $\chi_0 N$ are coarse-grained parameters. The spatial discretization and the presence of density fluctuations give rise to deviations of the inverse compressibility κN , the Flory-Huggins parameter χN and the mean squared end-to-end distance R_e^2 from the corresponding model parameters. For most systems, these deviations are small. For this reason, the physical quantities and model parameters will be used interchangeably in the following.

Instead of using direct MC simulations, it is useful to look at typical scales of the bonded and non-bonded interactions. The bonded interactions are typically much stronger and less computationally expensive. For a given χN , the typical ratio of bonded to non-bonded forces scales like $N^{3/2}$ [21]. Furthermore, the variations of the non-bonded interactions are much slower. The key idea behind SCMF simulations is to exploit these scaling differences to replace the explicit non-bonded interactions by quasi-instantaneous external fields $\omega_{\alpha,m}$:

$$\begin{aligned} \omega_{\alpha,m} &= \frac{1}{k_B T \rho_0 \Delta L^3} \frac{\partial \mathcal{H}_{\text{nb}}}{\partial \hat{\phi}_{\alpha,m}}, \\ \omega_{A,m} &= \kappa N (\hat{\phi}_{A,m} + \hat{\phi}_{B,m} - 1) - \frac{\chi N}{2} (\hat{\phi}_{A,m} - \hat{\phi}_{B,m}), \end{aligned} \quad (2.22)$$

and similarly for $\omega_{B,m}$. This way, Equation 2.21 is rewritten to:

$$\frac{\mathcal{H}_{\text{nb}}^{\text{SCMF}}[\hat{\phi}_A, \hat{\phi}_B]}{k_B T} = \frac{\rho_0 \Delta L^3}{2N} \sum_{m=1}^{N_{\text{cells}}} \omega_{A,m} \hat{\phi}_{A,m} + \omega_{B,m} \hat{\phi}_{B,m}. \quad (2.23)$$

The actual SCMF algorithm consists of two steps. First, the external fields are computed according to Equation 2.22. Subsequently, they are kept constant over a short duration during which a MC scheme is used to update the particle positions. A trial move is proposed according to a predefined rule and accepted according to the Metropolis-Hastings criterion [22]:

$$p_{\text{acc}} = \min \left\{ 1, \exp \left(-\frac{\Delta E}{k_B T} \right) \right\}, \quad (2.24)$$

where $\Delta E = \Delta \mathcal{H}_b + \Delta \mathcal{H}_{\text{nb}}^{\text{SCMF}}$ is the energy change associated with the move. After a predefined number of MC steps, the external fields are recalculated from the current densities and the procedure is repeated. The independent time evolution of polymers thanks to the quasi-instantaneous field approximation allow for an efficient implementation on parallel computers. The accuracy of the SCMF algorithm depends on the update frequency of the external fields and the parameter

$$\epsilon = \frac{1}{\sqrt{N \bar{N}}} \left(\frac{b_0}{\Delta L} \right)^3, \quad (2.25)$$

which needs to be small [5].

2.5 Smart Monte Carlo Move

Traditionally, a MC trial move would consist of a displacement of each particle by a random, normalized, Gaussian distributed vector $\Delta \mathbf{R}$. The freedom of choice of this trial move in the original Metropolis Algorithm [22] can be exploited to achieve higher acceptance rates. The Smart Monte-Carlo (SMC) move introduces an additional term proportional to the bonded forces $\mathbf{F}_i(\mathbf{r})$ [23, 24]:

$$\mathbf{r}'_i(s) = \mathbf{r}_i(s) + \mathbf{F}_i(\mathbf{r}) \Delta A + \Delta \mathbf{R}, \quad (2.26)$$

which is simply and Euler integration with time step $\Delta t = \zeta_0 \Delta A$.

2.6 Discrete optimization

As will be shown in chapter 6, applying non-periodic boundary conditions in SCMF simulations by converting molecule types leads to a discrete optimization problem. This type of problem involves finding the best configuration S^* out of a finite set Ω of possible configurations, e.g., the one that minimizes a certain loss function $f(S)$, where S is an arbitrary configuration. Ω is often called *search space*. In this thesis, only minimization problems are considered. Maximization is completely analogous. These problems are often NP-complete [25], and, therefore, designing efficient algorithms is very difficult. One usually distinguishes between convex and non-convex optimization problems. Convex problems only have a single, global minimum. The optimal solution can be found efficiently by applying a greedy algorithm. In each iteration step, a new solution is proposed and accepted only if it leads to a reduction in the loss function [26]. It therefore gradually moves towards the global optimum. If the problem is non-convex, then there exist multiple local minima in which the greedy algorithm is likely to become trapped.

2.7 Simulated Annealing

SA [27] is a heuristic algorithm that is frequently used to find at least an approximate optimum for a non-convex discrete optimization problem. It is inspired by the annealing process in metallurgy, in which the crystallization of a material is enhanced by heating it up to a high temperature and letting it cool down slowly [28]. For this, a temperature parameter T is introduced. It starts from an initially high temperature T_0 which is reduced according to a predefined rule

$$T_k = g(k), \quad (2.27)$$

until a minimal temperature T_f is reached. Initially, the algorithm makes an initial guess $S_0 \in \Omega$ for the optimal solution. At each temperature, a new solution S^* is created from the current solution S according to a specific generation rule. The new solution is accepted with a probability given by the Metropolis-Hastings

2 Theory

criterion, similar to Equation 2.24:

$$p(S \rightarrow S^*) = \min \left\{ 1, \exp \left(-\frac{f(S^*) - f(S)}{T} \right) \right\}. \quad (2.28)$$

Instead of proposing only a single neighboring solution at each temperature, multiple trials are also possible. This is essentially the same as performing a MC simulation while lowering the temperature. Much like atoms in a solid have larger degrees of freedom at high temperatures, new solutions have a higher probability of being accepted at high temperatures. Accepting worse solutions leads to a good *exploration* of the search space and it can prevent the algorithm from becoming trapped in a local minimum. As the temperature is decreased, the algorithm transitions towards an *exploitation* phase, in which mostly only better solutions are accepted. The efficiency of SA and the quality of the optimum depends greatly on the applied cooling schedule. Under a certain set of conditions, the algorithm is guaranteed to converge if the cooling is logarithmic [29], e.g.,

$$T_k = \frac{d^*}{\log(k+1)}, \quad (2.29)$$

where d^* is the maximum depth of all local minima excluding the global minimum. This theoretical result is not very useful, since d^* is difficult to determine and logarithmic cooling is much too slow for practical use. The cooling schedule must be chosen in order to achieve a good trade-off between computational efficiency and quality of the optimum. There is no general choice for T_0 , T_f and $g(k)$ in Equation 2.27 that yields good results for any problem. Rather, the parameters have to be tuned empirically to the problem at hand. This is further complicated by the fact that, in most cases, no a priori guess can be made for the value of f at the global minimum.

3 Simulation technique

To simulate the collective dynamics, a coarse-grained model of the polymers is employed. Within this model, several monomeric repeat units are grouped into *beads*, which allows for an efficient numerical implementation. Nevertheless, in this study, the terms “bead” and “monomer” will be used interchangeably. Each macromolecule is assigned one of m_t possible architectures upon initialization. The architecture describes the bead types and bond topology of the macromolecule. Other than simple homopolymers, more complex bond topologies like copolymers, star polymers and rings are possible. A great variety of universal properties of polymeric materials on mesoscopic length scales is accurately captured by coarse-grained models [30].

The software package that is used for the numerical calculations, SOMA, uses a combination of the described coarse-grained model and the SCMF algorithm [6]. Unlike the widely used SCFT, the SCMF method includes fluctuation effects which are required to accurately describe certain systems and effects, e.g. dilute polymer solutions, the vicinity of phase transitions, or polymeric microemulsions [31–33]. The interactions are fully described by three coarse-grained parameters: the average mean squared end-to-end distance R_e^2 of a chain in the absence of non-bonded interactions, the inverse thermal compressibility $\kappa_0 N$ and the incompatibility between different bead types $\chi_0 N$. The term “soft” in SOMA relates to the soft nature of the non-bonded interactions, which arises from the systematic coarse-graining and allows for an overlap of beads [34]. As mentioned before, the enormous benefit of the SCMF algorithm is that the chains are decoupled, making it possible to implement it effectively on parallel machines. For this, the Message Passing Interface (MPI) standard is employed. Each molecule is assigned to a specific MPI rank and the densities are computed according to Equation 2.20. SOMA also supports parallelization on shared-memory systems using the OpenMP standard as well as the use of accelerators like Nivida Graphics processing units (GPUs) [5] using the OpenACC standard. For the particle propagation, a SMC scheme is employed that uses the strong bonded forces to propose a trial displacement resembling Brownian motion

3 Simulation technique

and produces Rouse-like dynamics [35, 36]. Another important feature of SOMA is the conversion of macromolecules to different types of the same bond architecture, but with different bead types. This allows for the simulation of chemical reactions, or, as used in this thesis, to generate a flux.

The conversion-based-target-composition plugin is implemented to read in target composition fields and optimize the polymer type configurations, as explained in chapter 6. The identification of the target polymers, as well as the matrices \tilde{q} and \tilde{n} , as explained in Algorithms 1-3, can be performed in parallel using OpenACC. The actual optimization process is inherently sequential and parallelization is thus impossible. Based on the initial temperature T_0 , an optimization algorithm is chosen: if $T_0 > 0$, SA will be performed. Otherwise, the SD algorithm is employed. The interaction part of the quasi-instantaneous external fields is modified as explained in section 6.6

4 Collective diffusion of symmetric homopolymers

To describe the collective diffusion behavior of polymer mixtures, the most important quantity is the Onsager coefficient. In this section, the simple case of an effectively one-dimensional current is investigated by simulation of a binary homopolymer mixture in a confined simulation box. The form Equation 2.17 for the Onsager coefficient is obtained.

4.1 Reference system

The collective diffusion properties of non-interacting homopolymers with $N_A = N_B = N$ and $\chi = 0$ are investigated. As a reference system, a simulation box with $n_A = n_B = 5000$ polymers and dimensions $L_x \times L_y \times L_z = 9.75 \times 3 \times 3 R_e^3$ is used, so the invariant degree of polymerization is $\sqrt{N} \approx 114$. The spatial discretization is $\Delta L = 0.125 R_e$. Periodic boundary conditions are applied in the lateral y and z directions, whereas impenetrable walls are applied in the x direction. Initially, the polymers are distributed homogeneously in the system. To stimulate a flux in a Non-equilibrium steady state (NESS), conversion zones of width $d = 0.25 R_e$ are introduced close to the walls. In each time step, if the center-of-mass coordinate \mathbf{r}_{cm} of a polymer of type A lies in the left conversion zone, it is converted to type B with probability $p(A \rightarrow B) = r$. Analogously, conversion from B to A takes place in the right conversion zone with the same probability r . The current \mathbf{J} is measured by tracking the number of polymer conversions. The simulation setup is depicted in Figure 4.2.

The computation of the transport properties is complicated by boundary effects, such as a steep density drop close to the hard walls, which is of entropic origin. As a result, the densities defined in Equation 2.20 do not exactly sum up to one in the bulk. In Figure 4.1, it can be seen that the total density is about 1.01 in the bulk

4 Collective diffusion of symmetric homopolymers

and drops to 0.65 in the cells that border the walls directly.

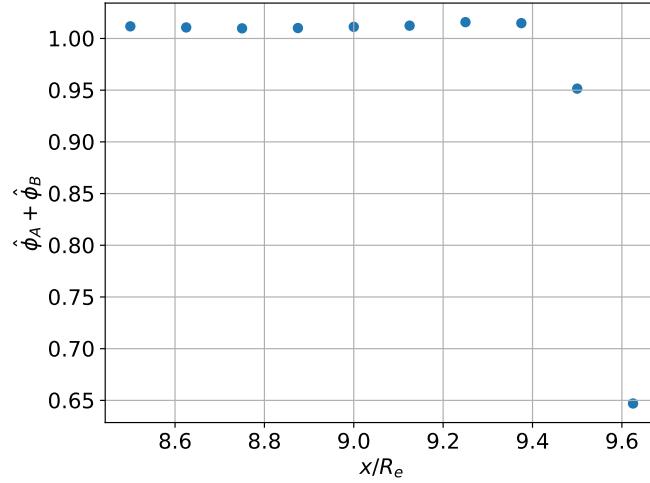


Figure 4.1: One dimensional profile of the total density.

Furthermore, chains whose center of mass lies in the conversion zone may extend far beyond that zone. The range of these effects is approximated as R_e and measurements are only taken in the region where the effects are negligible. The total densities are re-normalized by dividing by the average value in the bulk.

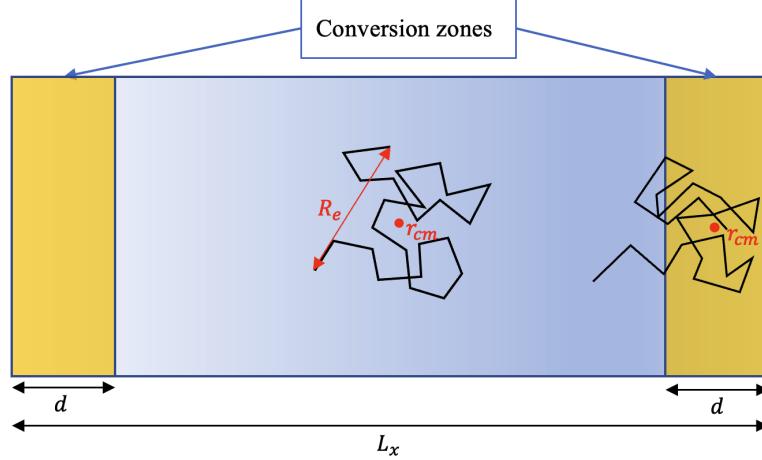


Figure 4.2: Simulation setup. For clarity, the conversion zones are not to scale and no distinction between polymer types is made.

4.2 Collective diffusion coefficient

Due to the periodic boundary conditions and the spatial homogeneity in the lateral directions, the system can effectively be described in one dimension. The density profile for $r = 1.0$ is shown in Figure 4.3.

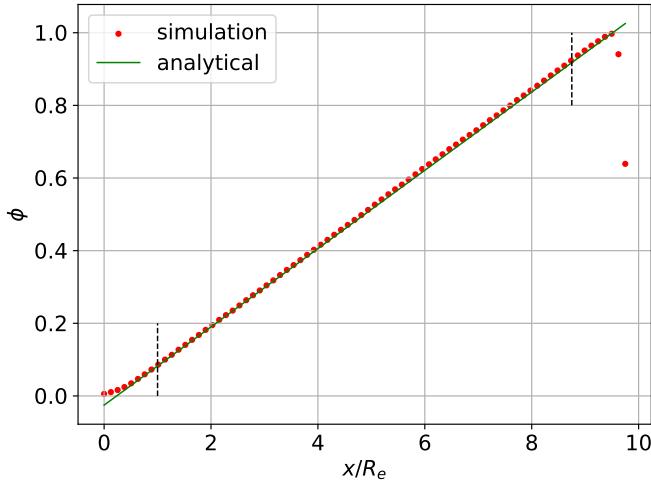


Figure 4.3: Steady-state density profile for $r = 1.0$ averaged over y and z . The analytical curve corresponds to Equation 4.5. The black lines mark the region that is estimated to be affected by boundary effects.

Outside of the range of the boundary effects, the density profile is very well represented by a linear function. The chemical potential is obtained by taking the functional derivative $\frac{\delta F}{\delta \phi}$ of Equation 2.9. Since $\chi = 0$, no phase separation occurs and the local density differences are entirely due to the dynamics. Assuming the WSL, the chemical potential becomes:

$$\begin{aligned} \frac{\mu R_e^3}{\sqrt{N} k_B T} &= \ln \phi - \ln(1 - \phi) - \frac{R_e^2}{18\phi(1 - \phi)} \phi'' \\ &\quad + \left[\frac{R_e^2(1 - 2\phi)}{36\phi^2(1 - \phi)^2} \right] \phi'^2, \end{aligned} \quad (4.1)$$

where the dashes denote derivatives with respect to x . The resulting chemical potential profile is shown in Figure 4.4a.

Evidently, the contribution of the terms involving spatial derivatives of the density to the total chemical potential is negligible, except in a very small region close to the walls. The linearity of the density profile already implies $\phi'' = 0$ and Figure 4.4

4 Collective diffusion of symmetric homopolymers

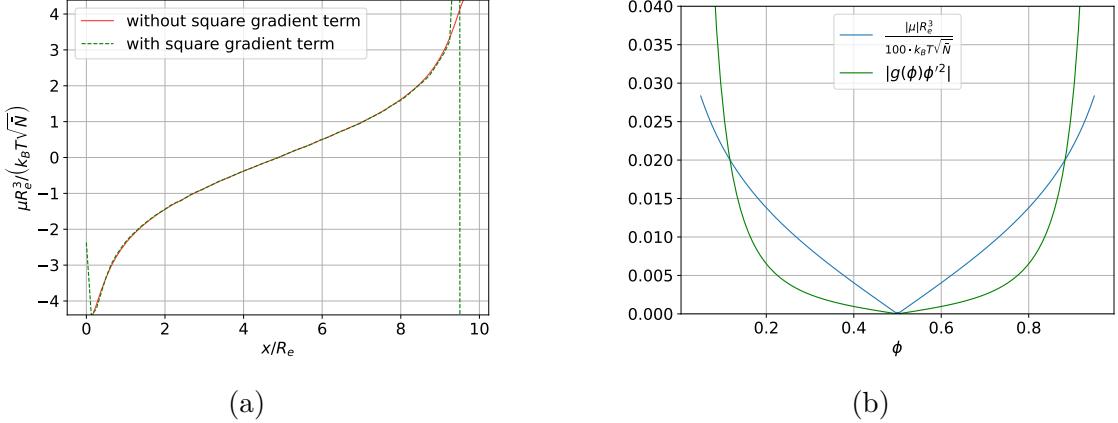


Figure 4.4: (a) Steady-state chemical potential profile obtained from Equation 4.1 for $r = 1.0$, averaged over y and z . (b) $g(\phi)\phi'^2 \equiv R_e^2(1-2\phi)\phi'^2/[36\phi^2(1-\phi)^2]$ and absolute value of chemical potential $|\mu|R_e^3/(100 \cdot k_B T \sqrt{\bar{N}})$. In the discussed case, $\phi' \approx 0.1$.

shows that the term proportional to ϕ'^2 accounts for less than one percent of the total chemical potential for $0.12 \lesssim \phi \lesssim 0.88$, which roughly corresponds to the region in Figure 4.3 that is assumed to be unaffected by boundary effects. In the subsequent discussion, the terms in Equation 4.1 containing spatial derivatives of ϕ will be neglected. This approximation improves further as the simulation box becomes infinitely long and $\phi' \rightarrow 0$. However, for more complex density profiles, the contribution is of course not negligible. The gradient of the chemical potential becomes:

$$\frac{\nabla \mu R_e^3}{\sqrt{\bar{N}} k_B T} = \frac{\phi'}{\phi(1-\phi)}. \quad (4.2)$$

With Equations (2.13) and (4.1), neglecting the terms arising from the square gradient, the current calculates to:

$$J = -D \sqrt{\bar{N}} R_e^{-3} \phi', \quad (4.3)$$

If the current of the normalized density is considered instead, Equation 4.3 must be divided by $\sqrt{\bar{N}} R_e^{-3}$. Together with Equation 2.11, this gives the standard diffusion

equation:

$$\frac{\partial\phi(\mathbf{r},t)}{\partial t} - D\phi'' = 0. \quad (4.4)$$

where D is obtained from Equation 2.6. In the steady state, this simply yields $\phi'' = 0$, so a linear density profile is obtained, as expected. From Equation 4.3, and the condition that $\phi(L_x/2) = 1/2$, which follows from the symmetry of the conversion rates, the density profile becomes:

$$\phi(x) = \frac{JR_e^3}{D\sqrt{N}} \left(\frac{L_x}{2} - x \right) + \frac{1}{2}, \quad (4.5)$$

To verify Equation 2.17, the Onsager coefficient may also be obtained directly from the simulation results using Equation 2.13. Again assuming local coupling and making use of Equation 4.2, one obtains:

$$\Lambda = -\frac{JR_e^3}{\sqrt{N}\phi'} \phi(1-\phi). \quad (4.6)$$

The resulting Onsager coefficient is plotted in Figure 4.5.

The Onsager coefficients obtained from Equations (2.17) and (4.6) are in excellent agreement, which hints that the theoretical derivations are consistent with the simulation results. Specifically, the assumptions of incompressibility and local coupling of the current to the chemical potential are justified and the linear density profile observed in Figure 4.3 is explained.

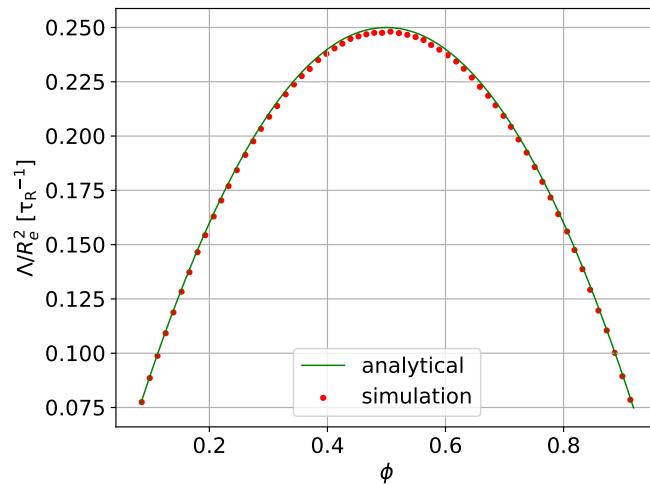


Figure 4.5: Onsager coefficient Λ as a function of ϕ . The analytical curve is obtained from Equations (2.17) and (4.5), the numerical curve from Equation 4.6 and the density profile in Figure 4.3, where ϕ' is obtained from a linear fit. Only data points measured at a distance greater than R_e from the walls are used. The diffusion constant D and the current J are both obtained from the simulation.

5 External-field-driven flow

One of the simplest ways to implement time-dependent boundary conditions in SOMA is to apply time-dependent external fields that are only nonzero close to the boundaries. For this, an additional, dimensionless term $f_i(\mathbf{r}, t)$ is added to the instantaneous fields Equation 2.22 which quantifies the excess free energy of a bead of type i in the external field in units of $k_B T$. In this section, layers of lamellar diblock-copolymers are moved at constant speed v perpendicular to their orientation using spatially periodic external fields close to the boundaries. In the applied MC scheme, the friction coefficient ζ in Equation 2.5 is determined relative to a fixed background. In a system that moves along with the external field, this background is moving at the speed $-v$, so the system is not Galilei-invariant and each monomer experiences a friction force $-\zeta v$, causing the whole lamellae to bend. The convective motion due to the moving fields is opposed by the diffusive motion due to the Brownian dynamics. It is therefore useful to define a Péclet number [37]:

$$P_e = vR_e/D, \quad (5.1)$$

which puts the two transport mechanisms in relation on the typical length scale R_e . For small $P_e \approx 0$, the chains are able to fully relax to the undeformed shape. Beyond a critical value P_e^* , they cannot keep up with the movement of the external field anymore, which causes the lamellae to break and reconnect to the boundaries periodically. This is shown in Figure 5.1 for $P_e = 0.92$, the whole process happens within about half a relaxation time. In this section, an intermediate regime of P_e is investigated to obtain the bending modulus K of the lamellae.

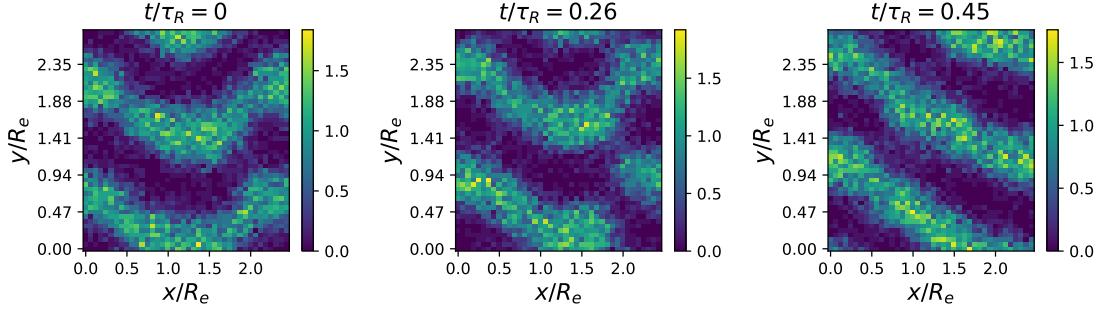


Figure 5.1: Snapshots of the A density at $P_e = 0.92$. Left: Shortly before breaking. Middle: Lamella breaks. Right: Broken end reconnects to nearest lamella part.

5.1 Reference system

A system of $n = 750$ symmetric diblock-copolymers with $\chi N = 20$ is used. This corresponds to a lamellar phase in Figure 2.2. The chain discretization is $N = 32$ or $N = 64$, respectively. The box dimensions are $L_x \times L_y \times L_z = 2.5 \times 2.82 \times 1 R_e^3$, which corresponds to $\sqrt{N} = 106$. The spatial discretizations are $\Delta x = 1/16 R_e$, $\Delta y = 47/800 R_e$ and $\Delta z = 1 R_e$. To generate the initial lamellar structure, external fields are applied, as shown in Figure 5.2. The interlayer spacing of $\lambda = 1.41 R_e$ was found to be stable throughout the simulations, but it does not correspond to the equilibrium spacing.

Subsequently, the external fields are switched off everywhere except at a distance less than $b = 0.5 R_e$ from the boundaries in the x -direction, so the length of the part of the lamellae that is not supported by the external fields is $L = 1.5 R_e$. After a time Δt , the fields are moved by a distance of Δy in the y -direction, so the velocity is $v = 47R_e/(800\Delta t)$. Snapshots of the density profile are also periodically taken after a time Δt . The external fields balance the friction forces at the boundaries and therefore act as bearings for the lamellae. The diffusion constant D is obtained from Equation 2.6, where $g_3(t)$ is measured in a system without external fields and $\chi N = 0$.

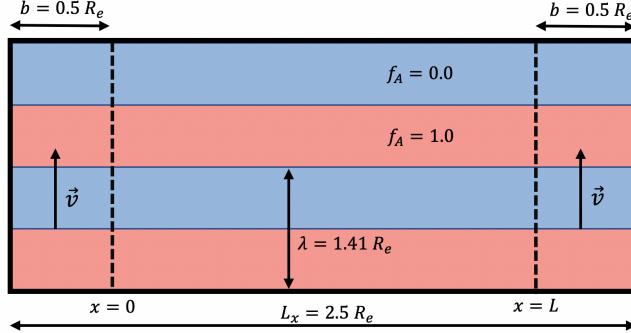


Figure 5.2: Sketch of the external field $f_A(\mathbf{r}, 0)$. Red domains correspond to $f_A = 1.0$, blue domains to $f_A = 0.0$. $f_B(\mathbf{r}, 0)$ is exactly complementary. Within the region bounded by the dotted lines, the external fields are switched off after the initial lamella structure has been generated.

5.2 Bending modulus

Neglecting any change in the lamellar spacing, in the SSL, the free energy of a bent lamella is [38]:

$$F = \int d\mathbf{r} \left\{ f_0 + \frac{1}{2} K (\partial_x^2 u)^2 \right\}, \quad (5.2)$$

$$\begin{aligned} &\approx \lambda L_z \int dx \left\{ f_0 + \frac{1}{2} K (\partial_x^2 u)^2 \right\} \\ &\equiv \lambda L_z \int dx f(u''). \end{aligned} \quad (5.3)$$

where f_0 is the free energy per unit volume of the unbent lamella, K is the bending modulus and $u \equiv u(x)$ is the deformation profile of the A-B-interface. Due to the bending, the lamellae also experience a volume change which is neglected in the integration. The friction force density is $q_{fric} = \rho_0 \zeta v = N \sqrt{N} \zeta P_e D / R_e^4$. In the steady-state, it is balanced by a force density given by the functional derivative $\delta F / \delta u$:

$$\begin{aligned} \frac{\delta F}{\delta u} &= \frac{\partial^2}{\partial x^2} \frac{\partial f}{\partial u''} \stackrel{!}{=} q_{fric} \\ \Rightarrow K \frac{\partial^4 u}{\partial x^4} &= N \sqrt{N} \zeta \frac{P_e D}{R_e^4}. \end{aligned} \quad (5.4)$$

With the boundary conditions $u(0) = u(L) = 0$ and $u''(0) = u''(L) = 0$, one

obtains:

$$u(x) = \frac{N\sqrt{N}\zeta P_e D}{24K R_e^4} x(L^3 - 2Lx^2 + x^3), \quad (5.5)$$

in analogy to a beam bending under a uniform load in the Euler-Bernoulli theory. The resulting lamellar profiles are shown in Figure 5.3 for $N = 32$. Due to the periodic boundary conditions, the zero point of the y axis can be shifted arbitrarily. This is exploited such that a whole A -lamella can be displayed in the lower half of the coordinate system, e.g., for $y < \lambda = l_y/2$. Figure 5.3a shows the average of both A -lamella in the system. Since the boundaries between two adjacent monolayers are not sharp, the “center of mass curve” is used instead to describe the one dimensional lamella profile. The deformation curve $u(x)$ is obtained as

$$u(x) = \left| \frac{1}{\hat{\phi}_A(x)} \left(\sum_{i=0}^{n_y/2-1} i\Delta y \hat{\phi}_A(x, i\Delta y) \right) - y_0 \right|, \quad (5.6)$$

where $\hat{\phi}_A(x) = \sum_{i=0}^{n_y/2-1} \hat{\phi}_A(x, i\Delta y)$ is the total A -density as a function of x . $n_y = l_y/\Delta y$ is the number of grid points in the y direction and y_0 is the center of mass of the A -block at $x = 0$, as indicated in Figure 5.3a. Equation 5.6 assumes that only a single A -block is considered, without interference from neighboring A -blocks. Figure 5.3b shows the fit of Equation 5.6 to Equation 5.5. For $P_e = 0.24$, the fit is in excellent agreement with the simulation data. $P_e = 0.85 \approx P_e^*$ was found to be the critical Péclet number beyond which the lamellae break. For these large deformations, the fit displays deviations from the simulation data, especially around the maximum. This error is systematic, since, for large deformations, the vertical extension of a single A -block is approximately λ , and, therefore, A -blocks from the neighboring bilayer interfere with the computation of Equation 5.6.

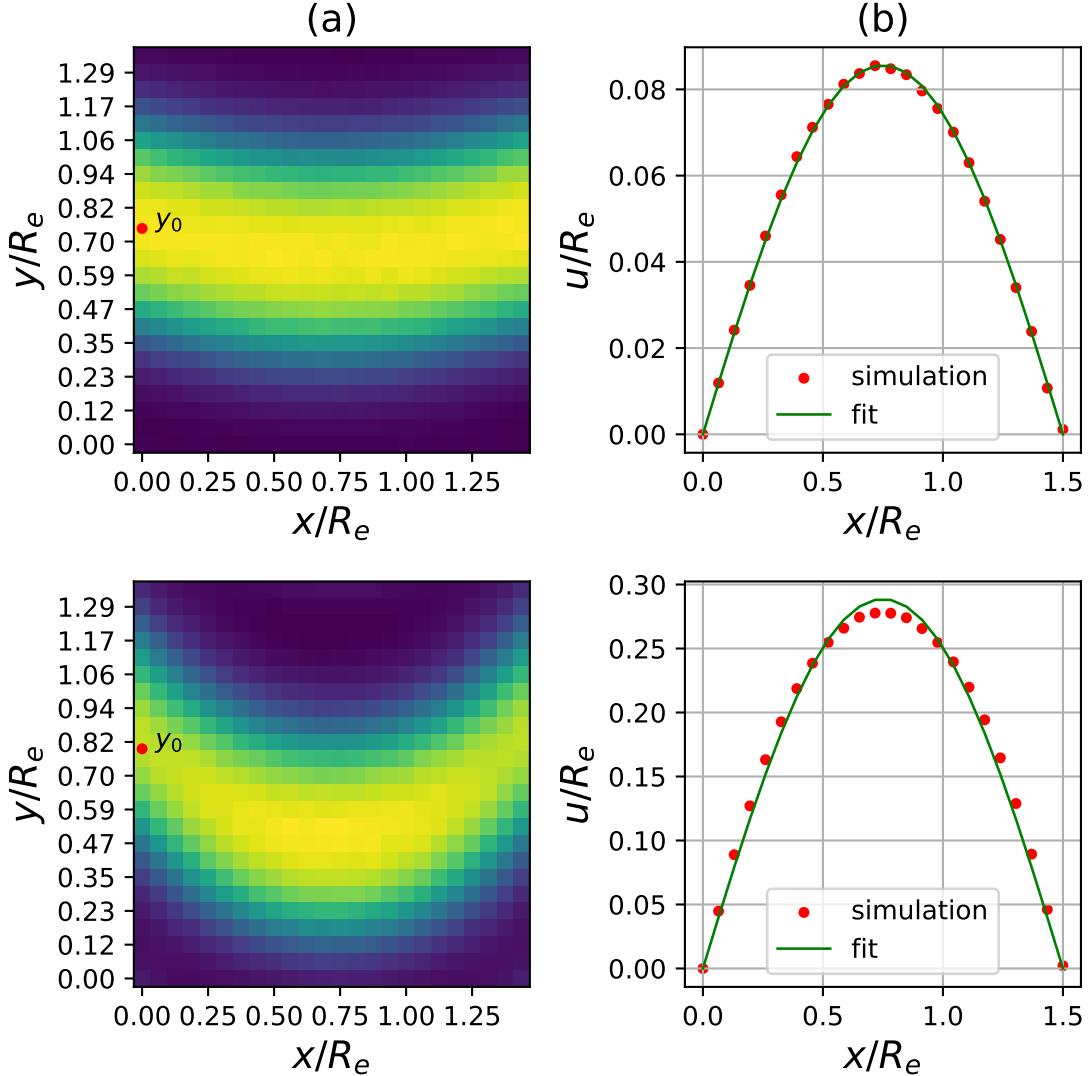


Figure 5.3: (a) Heatmap of the steady-state lamella profile in the reference frame that moves with the external field, averaged over all lamellae. (b) Lamella deformation curve obtained from Equation 5.6. The fit corresponds to Equation 5.5. The top row corresponds to $P_e = 0.24$. The bottom row corresponds to $P_e = 0.85$.

The maximum deformation is:

$$u_{max} = u(L/2) = \frac{5N\sqrt{N}\zeta P_e DL^4}{384KR_e^4}. \quad (5.7)$$

To obtain the bending modulus, u_{max} is measured for various values of P_e , this is shown in Figure 5.4a. Values corresponding to $P_e > 0.7$ are discarded due to the

5 External-field-driven flow

systematic error for large deformations. Figure 5.4b shows that the bending modulus is proportional to $1/N$.

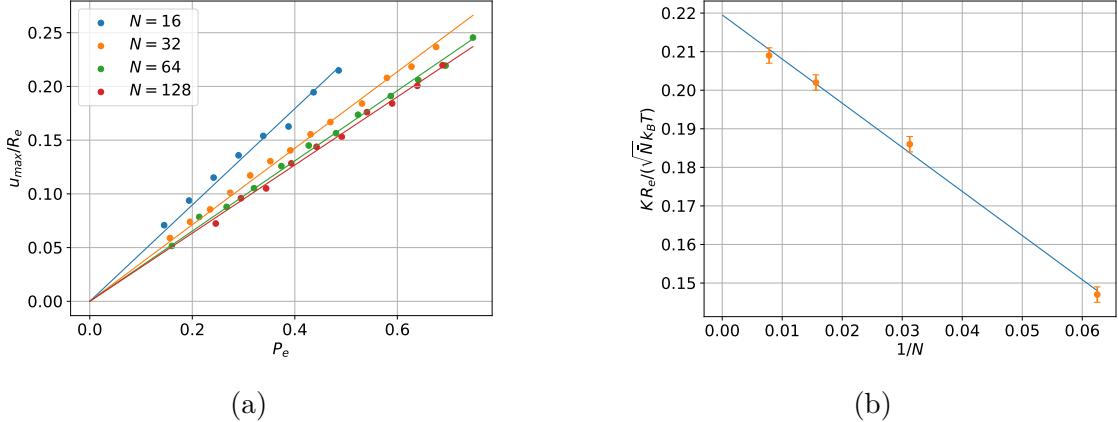


Figure 5.4: (a) Maximum deformation u_{max} as a function of the Péclet number P_e . The fits correspond to Equation 5.7. (b) Bending modulus K plotted over $1/N$.

Alternatively, K may also be obtained from SCFT. While a detailed discussion of SCFT is omitted in this thesis, it should be mentioned that, unlike in the particle-based SCMF simulations, the chains are treated as continuous density fields with $N \rightarrow \infty$. The system is forced away from equilibrium by constraining the A - B -interface to a sinusoidal deformation $u(x) = a \sin 2\pi x/L$, as shown in the inset of Figure 5.5. From Equation 5.3, the total free energy may be calculated as:

$$\begin{aligned} F &\approx \lambda L_z \int dx \left\{ f_0 + \frac{1}{2} K a^2 \left(\frac{2\pi}{L} \right)^4 \sin^2 \left(\frac{2\pi}{L} x \right) \right\} \\ &= \lambda L L_z f_0 + \lambda L_z \frac{4\pi^4 a^2}{L^3} K. \end{aligned} \quad (5.8)$$

On the other hand, the total free energy may be obtained by integrating the free energy density over the lamella volume:

$$\begin{aligned} F &= \int d\mathbf{r} (f_0 + f_b) \\ &\approx \lambda L L_z (f_0 + f_b) \\ &= \lambda L L_z \frac{\sqrt{N}}{R_e^3} (\tilde{f}_0 + \tilde{f}_b), \end{aligned} \quad (5.9)$$

5 External-field-driven flow

where $\tilde{f}_0 = 4.0337 k_B T$ is the free energy per chain of the undeformed lamella and $\tilde{f}_b = 0.79278 k_B T (a/R_e)^2$ is the free energy per chain due to the deformation. Comparing Equations (5.8) and (5.9), the bending modulus reads:

$$K_{\text{SCFT}} = \frac{\tilde{f}_b \sqrt{N}}{4a^2 \pi^4} R_e^{-3} L^4. \quad (5.10)$$

The result is $K_{\text{SCFT}} = 0.16 \sqrt{N} k_B T / R_e$. Extrapolation from the linear fit in Figure 5.4b to $N \rightarrow \infty$ gives $K_\infty = 0.22 \sqrt{N} k_B T / R_e$. Furthermore, the difference may arise from the different mechanisms that are used to keep the systems away from equilibrium: the moving external field in the SCMF simulation and the interfacial constraint in the SCFT simulations.

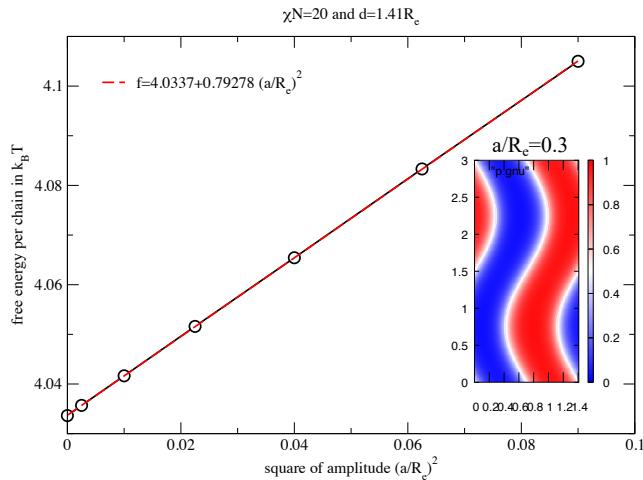


Figure 5.5: Free energy per chain as a function of the square of the amplitude from exact SCFT calculations. The inset shows the lamella cross-section for $a/R_e = 0.3$.

6 Polymer-type-conversion-based target composition

6.1 Formulation of the problem

We now want to take a step towards simulating the setting depicted in Figure 1.1. If one considers two opposing sides from the rectangular section cut out of the larger continuum simulation, then the time-evolution of the density will, in general, be completely different on both walls. Therefore, periodic boundary conditions cannot be applied. Instead, the simulation box is bounded by hard, impenetrable walls in SOMA. As discussed previously, the proximity area of these walls is entropically disfavored, leading to a decrease in the total density. This is problematic, since it is exactly this region where one wishes to dictate a target composition. Another issue is caused by the fluctuation of the total density, which is one of the key features of SCMF simulations. A way out is to dictate the composition instead:

$$\tilde{\phi}_\alpha(c) = \frac{\hat{\phi}_\alpha(c)}{\sum_{i=1}^{n_t} \hat{\phi}_i(c)}. \quad (6.1)$$

It brings the advantage that the compositions of different types always sums to one. However, it is undefined for empty cells, so these must be ignored. To dictate the composition in specific cells, target values $\tilde{\phi}_{\alpha,T}(c)$ are defined, with $\tilde{\phi}_{\alpha,T}(c) = -1 \forall \alpha$, a purposefully unphysical value, for cells in which no target composition is desired. In the following, the cells c for which $\tilde{\phi}_{\alpha,T}(c) \geq 0 \forall \alpha$ are called “target cells”. Mixed cells with $\tilde{\phi}_{\alpha,T}(c) \geq 0$ for some α and $\tilde{\phi}_{\alpha,T}(c) < 0$ for other α are not allowed and target cells must fulfill the condition $\sum_\alpha \tilde{\phi}_{\alpha,T}(c) = 1$. The total number of target cells is denoted n_T . Let n_p be the number of polymers that have at least one monomer in any target cell c . In the following, these polymers are called “target polymers”. Instead of using external fields, like in the previous section, the goal is

to reach the target composition by switching polymer types, see chapter 4. In this thesis, only conversions between macromolecule types with the same bond topology are considered. Due to the chain connectivity, this leads to a complex optimization problem, which aims to minimize the following loss function:

$$L[\{\hat{\phi}_i\}] = \sum_{\alpha=0}^{n_t} \sum_{c \in \{\text{cells}\}} \theta(\tilde{\phi}_{\alpha,T}(c)) (\tilde{\phi}_{\alpha}(c) - \tilde{\phi}_{\alpha,T}(c))^2, \quad (6.2)$$

where $\theta(\tilde{\phi}_{\alpha,T}(c)) = 1$ if $\tilde{\phi}_{\alpha,T}(c) \geq 0$ and $\theta(\tilde{\phi}_{\alpha,T}(c)) = 0$ otherwise. Since the compositions $\tilde{\phi}_{\alpha}(c)$ depend on the target polymer types, the search space Ω consists of all possible combinations of target polymer types. To minimize Equation 6.2, one must first define the notion of a neighboring solution. If $\beta = \{\beta_1, \dots, \beta_j, \dots, \beta_{n_p}\} \in \Omega$ is the current configuration of polymer types, then we define a neighboring solution as $\beta^* = \{\beta_1, \dots, \beta_j^*, \dots, \beta_{n_p}\} \in \Omega$, where the type of the j th target polymer is changed from β_j to β_j^* and every other polymer type is unchanged. In the following, this rule to generate a neighboring solution is called a “flip”. Equation 6.2, together with the flip rule, defines a non-convex discrete optimization problem. The non-convexity can easily be seen by looking at the simple example for $n = 4$ homopolymers with $N = 2$ on a 2×2 grid shown in Figure 6.1.

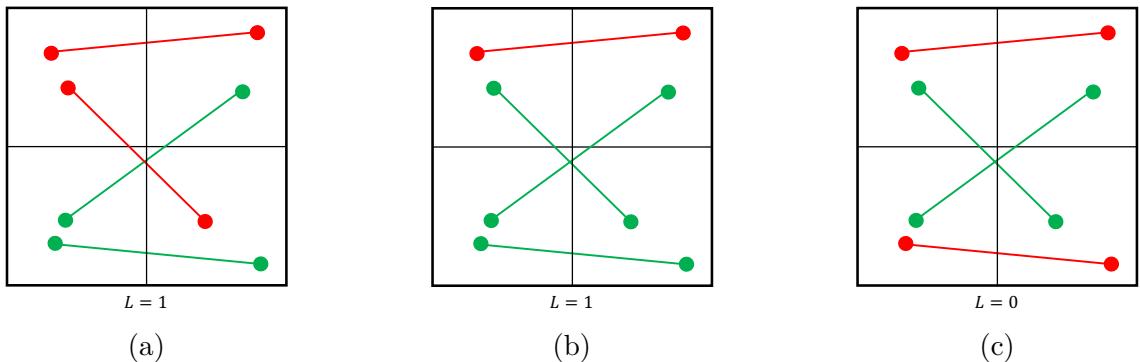


Figure 6.1: Example for $n = 4$ homopolymers with $N = 2$ on a 2×2 grid to show that the loss function defined by Equation 6.2 is, in general, non-convex under the type-switch of a single polymer. The target compositions are $\tilde{\phi}_{A,T} = \tilde{\phi}_{B,T} = 0.5$ for all cells.

In Figure 6.1a, any possible flip will not decrease the value of L . Figure 6.1b shows the configuration after conversion of a red polymer to a green one. A flip of the polymer at the bottom now decreases the loss function, as shown in Figure 6.1c.

It should be noted that this minimum is not unique, since a flip of every polymer would also yield $L = 0$.

The target polymers are identified and saved to a list by Algorithm 1.

Algorithm 1 Get target polymers

```

1: Let  $polyIsTarget[n]$  be a new boolean array initialised with 0's
2:  $n_p \leftarrow 0$ 
3: for  $poly \leftarrow 0$  to  $n - 1$  do
4:   for  $monomer \leftarrow 0$  to  $N - 1$  do
5:      $monoCell \leftarrow$  cell index of monomer
6:     if target composition available for  $monoCell$  then
7:        $polyIsTarget[poly] \leftarrow 1$ 
8:        $n_p \leftarrow n_p + 1$ 
9:       break
10:      end if
11:    end for
12:  end for
13: Let  $targetPolyIndices[n_p]$  be a new array
14:  $i \leftarrow 0$ 
15: for  $poly \leftarrow 0$  to  $n - 1$  do            $\triangleright$  Store indices of target polymer to new array
16:   if  $polyIsTarget[poly] == 1$  then
17:      $targetPolyIndices[j] \leftarrow poly$ 
18:      $j \leftarrow j + 1$ 
19:   end if
20: end for

```

A boolean array of length n is declared and all values are set to zero. An iteration over all polymers and the corresponding monomers is performed. The monomers are assigned to a cell and it is checked if a target composition is available for that cell. If so, the corresponding entry for the polymer in the boolean array is set to one and the next polymer is considered. The number of target polymers n_p is also determined in this process. For a quicker access to the target polymers, their indices are stored sequentially in a separate array of length n_p . This requires dynamic memory allocation since n_p varies during the simulation and it cannot be determined analytically. The cell index of the k th target cell in which a target polymer j has monomers is saved to an $n_p \times N$ matrix \tilde{q} with entries \tilde{q}_{jk} . The number of target cells over which a target polymer j extends is denoted $n_{cells,j}$. It is not actually used in the implementation, however, it simplifies the discussion. $n_{cells,j}$ may take any integer value between 1 and N , this is incorporated by setting $\tilde{q}_{jk} < 0$ for $k \geq n_{cells,j}$. Let \tilde{n} be the $n_p \times m_t \times n_t \times N$ matrix with entries $\tilde{n}_{j\beta\alpha k}$ that holds the

6 Polymer-type-conversion-based target composition

number of monomers of type α that target polymer j has in the k th target cell if it has the molecule type β , where $\tilde{n}_{j\beta\alpha k} = 0$ for $k \geq n_{cells,j}$. An example is shown in Figure 6.2 for a binary mixture of homopolymers with $n_p = 2$, $N = 8$ and $n_T = 6$.

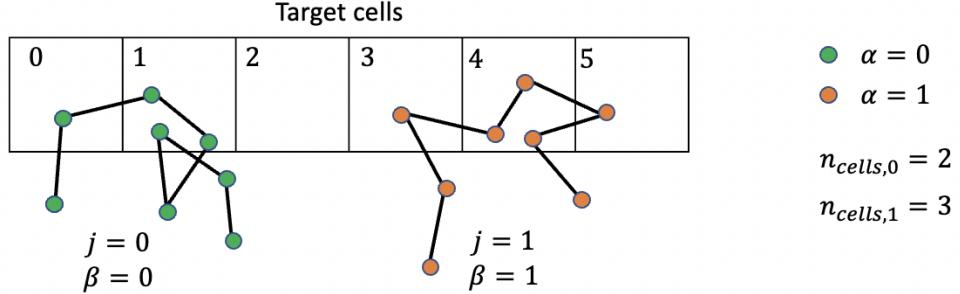


Figure 6.2: Example to explain the setting discussed in the text. β denotes the polymer type and α denotes the monomer type. For homopolymers, distinguishing between the two is redundant. For demonstration purposes, the differentiation is still made here.

The example corresponds to

$$\tilde{q} = \begin{bmatrix} 0 & 1 & -1 & -1 & -1 & -1 \\ 3 & 4 & 5 & -1 & -1 & -1 \end{bmatrix}$$

and the only non-zero submatrices

$$\tilde{n}_{j00k} = \tilde{n}_{j11k} \begin{bmatrix} 1 & 3 & 0 & 0 & 0 \\ 1 & 3 & 1 & 0 & 0 \end{bmatrix}.$$

of \tilde{n} . Any submatrices corresponding to $\alpha \neq \beta$ only have zero-entries since homopolymers are considered.

After the flip $\beta_j \rightarrow \beta_j^*$ of an arbitrary target polymer j , the value of the loss function is updated according to:

$$\begin{aligned}
 L_{\beta_j \rightarrow \beta_j^*}^*[\{\hat{\phi}_i\}] &= L[\{\hat{\phi}_i\}] - \sum_{\alpha=0}^{n_t} \sum_{k=0}^{n_{cells,j}-1} \left(\hat{\phi}_\alpha(\tilde{q}_{jk}) - \tilde{\phi}_{\alpha,T}(\tilde{q}_{jk}) \right)^2 \\
 &\quad + \sum_{\alpha=0}^{n_t} \sum_{k=0}^{n_{cells,j}-1} \left(\hat{\phi}_\alpha(\tilde{q}_{jk}) + \Delta\hat{\phi}_{j\alpha k}(\beta \rightarrow \beta^*) - \tilde{\phi}_{\alpha,T}(\tilde{q}_{jk}) \right)^2 \\
 &= L[\{\hat{\phi}_i\}] + \sum_{\alpha=0}^{n_t} \sum_{k=0}^{n_{cells,j}-1} \left\{ \Delta\hat{\phi}_{j\alpha k}^2(\beta \rightarrow \beta^*) \right. \\
 &\quad \left. + 2\Delta\hat{\phi}_{j\alpha k}(\beta \rightarrow \beta^*) \left(\hat{\phi}_\alpha(\tilde{q}_{jk}) - \tilde{\phi}_{\alpha,T}(\tilde{q}_{jk}) \right) \right\} \tag{6.3}
 \end{aligned}$$

where $\Delta\hat{\phi}_{j\alpha k}(\beta_j \rightarrow \beta_j^*) \equiv \tilde{n}_{j\beta_j^*\alpha k} - \tilde{n}_{j\beta_j\alpha k}$ is the change in the density of type α in cell \tilde{q}_{jk} due to the flip $\beta_j \rightarrow \beta_j^*$. Thus, for an efficient implementation of an optimization algorithm, the matrices \tilde{q} and \tilde{n} have to be determined first. For the determination of the unique target cells of a given target polymer, one first has to iterate again over all its bead positions and determine the corresponding cells using the assignment function in Equation 2.20. The cells are then sorted by their indices and the unique ones are identified. This is described in Algorithm 2.

Algorithm 2 Get unique target cells

```

1: Let  $targetPolyCells[n_p][N]$  be a new array initialised with  $-1$ 's
2: for  $i \leftarrow 0$  to  $n_p - 1$  do                                 $\triangleright$  Loop over target polymer
3:    $poly \leftarrow polymers[targetPolyIndices[i]]$ 
4:   let  $monoCells[N]$  be a new array
5:    $j \leftarrow 0$ 
6:   for  $monomer \leftarrow 0$  to  $N - 1$  do           $\triangleright$  Store all target cells to monoCells
7:      $monoCell \leftarrow cell\ index\ of\ monomer$ 
8:     if Target composition available for  $monoCell$  then
9:        $monoCells[j] \leftarrow monoCell$ 
10:       $j \leftarrow j + 1$ 
11:      break
12:    end if
13:   end for
14:   quicksort( $monoCells, 0, j - 1$ )
15:    $k \leftarrow 0$ 
16:   for  $monomer \leftarrow 0$  to  $j - 1$  do           $\triangleright$  Find unique cells
17:     if  $monoCells[monomer] \neq monoCells[monomer + 1]$  then
18:        $targetPolyCells[i][k] \leftarrow monoCells[monomer]$ 
19:        $k \leftarrow k + 1$ 
20:     end if
21:   end for
22:    $targetPolyCells[i][k] \leftarrow monoCells[j - 1]$ 
23: end for

```

This may seem tedious, however, it is necessary since a polymer can leave a given cell and reenter it at later segments, see Figure 6.2. For a homopolymer, the number of monomers it has of type α in a target cell c is simply given by the polymer type and the number of monomers it has in that cell. For a copolymer, it is necessary to compute the exact number of monomers it has of a given type in a given target cell, for each possible architecture it can assume. How this is done in SOMA is explained in Algorithm 3. The algorithm assumes that the matrix \tilde{q} has already been calculated.

Algorithm 3 Get monomer numbers

```

1: Let  $polyCellNum[n_p][m_t][n_t][N]$  be a new array initialized with 0's
2: for  $i \leftarrow 0$  to  $n_p - 1$  do                                 $\triangleright$  Loop over target polymer
3:    $poly \leftarrow polymers[targetPolyIndices[i]]$ 
4:    $initialPolytype \leftarrow poly.type$ 
5:   for  $polyType \leftarrow 0$  to  $m_t - 1$  do
6:      $poly.type \leftarrow polyType$                        $\triangleright$  Temporarily change polymer type
7:     for  $monomer \leftarrow 0$  to  $N - 1$  do
8:        $monoCell \leftarrow cell\ index\ of\ monomer$ 
9:        $monoType \leftarrow bead\ type\ of\ monomer$ 
10:      for  $j \leftarrow 0$  to  $N - 1$  do                   $\triangleright$  Find corresponding entry
11:        if  $targetPolyCells[i][j] == monoCell$  then
12:          Increment  $polyCellNum[i][polyType][monoType][j]$  by 1
13:          break
14:        end if
15:      end for
16:    end for
17:  end for
18:   $poly.type \leftarrow initialPolytype$                  $\triangleright$  Restore polymer type
19: end for

```

6.2 Steepest Descent

In the SD algorithm, which is a greedy algorithm, target polymers are selected at random and flipped to a random target type. If a flip decreases the loss function, it is accepted. Otherwise, it is rejected and the type of the according target polymer is reset to the previous type. This process is repeated until no more flips are accepted, out of n_p trials. Subsequently, one more flip trial is performed for each target polymer. This algorithm will get stuck in the first local minimum that it reaches.

6.3 Simulated annealing

To avoid becoming trapped in local minima, the SA algorithm described in section 2.7 is employed. On average, the algorithm attempts to flip each target polymer once at each temperature. The implementation is shown in Algorithm 4. The SD algorithm is run on the final configuration to ensure that at least a local minimum of Equation 6.2 is reached.

Algorithm 4 Simulated annealing

```

1:  $S \leftarrow$  current configuration of target polymer types
2:  $S_{\text{best}} \leftarrow S$ 
3:  $T \leftarrow T_0$ 
4: while  $T > T_f$  do
5:    $n_{\text{flip}} \leftarrow 0$ 
6:   while  $n_{\text{flip}} < n_{\text{max}}$  do
7:     generate new configuration  $S^*$  by flipping a random polymer
8:      $n_{\text{flip}} \leftarrow n_{\text{flip}} + 1$ 
9:      $\Delta L \leftarrow L(S^*) - L(S)$ 
10:     $r \leftarrow$  random number between 0 and 1
11:    if  $r < \exp(-\frac{\Delta L}{T})$  then
12:       $S \leftarrow S^*$ 
13:    end if
14:   end while
15:   if  $L(S) < L(S_{\text{best}})$  then
16:      $S_{\text{best}} \leftarrow S$                                  $\triangleright$  Update best solution
17:   end if
18:    $T \leftarrow T - \alpha$                              $\triangleright$  or different cooling schedule
19: end while

```

6.4 Comparison of algorithms

In order to compare the performance of the two presented algorithms, several randomly generated initial configurations are optimized with lamellar target compositions. A binary blend of homopolymers with $N = 32$ and $\sqrt{N} = 100$ is used. The box dimensions are $L_x \times L_y \times L_z = 5 \times 5 \times 5 R_e^3$ with $\Delta L = 1/8 R_e$ and hard, impenetrable walls are applied at all boundaries. The target composition is defined directly behind the walls with a width of one grid point. The lamellar target structure is a piecewise constant function that depends only on x and is characterized by the periodicity λ and the target deviation $\Delta\tilde{\phi}$ from the composition of the homogeneous value 0.5. The optimized structure obtained using the SD algorithm is shown in Figure 6.3 for $\lambda = 2 R_e$ and $\Delta\tilde{\phi} = 0.5$. The loss function is normalized by the number of monomer types n_t and the number of target cells n_T to obtain the mean squared deviation from the target composition. The thermal density variance σ_0^2 is taken as a reference value. It is calculated as $\sigma_0^2 = 1/\rho_0 \Delta L^3 \kappa$ in the mean field approximation [6]. Two different cooling schedules are investigated.

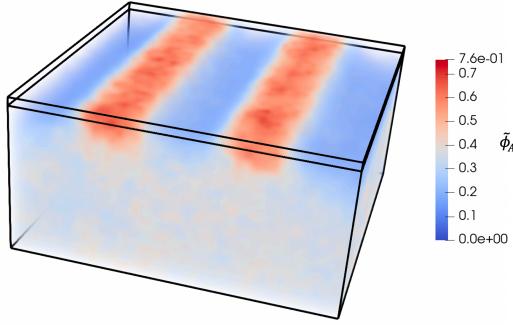


Figure 6.3: Composition $\tilde{\phi}_A(\mathbf{r})$ after optimization for $\lambda = 2 R_e$ and $\Delta\tilde{\phi} = 0.5$, obtained using SD algorithm. Averaged over 100 optimization runs starting from randomly generated initial configurations. The conversion zone is marked by the thin frame at the top.

6.4.1 SA parameter choice

As discussed previously, the performance of the SA algorithm depends greatly on the applied temperature reduction scheme. A commonly used scheme involves Linear Cooling (LC). The temperature in the k -th cycle is calculated according to

$$T_{k,LC} = T_0 - \alpha_{LC} k, \quad (6.4)$$

where $\alpha_{LC} > 0$. The total number of temperature cycles is $n_{LC} = \lceil (T_0 - T_{f,LC})/\alpha_{LC} \rceil$. While other cooling schedules might be more efficient, LC allows for a detailed exploration of the search space. Another option is to use an Exponential Cooling (EC) schedule:

$$T_{k,EC} = T_0 \alpha_{EC}^k, \quad (6.5)$$

where $0 < \alpha_{EC} < 1$. This corresponds to $n_{EC} = \lceil \log_{\alpha_{EC}}(T_f/T_0) \rceil$ temperature cycles. The minimum value of the loss function at a given temperature T for the different cooling schedules is denoted $L_{opt,i}(T)$, with $i = LC, EC$. This is shown in Figure 6.4a for a randomly generated initial configuration. The corresponding acceptance rate is shown in Figure 6.4b. The parameters are $\lambda = 1.25 R_e$, $\Delta\tilde{\phi} = 0.5$, $T_0 = 10$, $T_{f,LC} = 0$ and $T_{f,EC} = 10^{-4}$. As a reference, the SD result is also

6 Polymer-type-conversion-based target composition

shown. With both cooling schedules, the SD level is reached only for temperatures $T < 0.1 \ll T_0$. However, the acceptance rate already displays a significant sensitivity to temperature decrease at much larger temperature scales. The maximum temperature is therefore chosen appropriately. The final value $L_{opt,LC}(0)/n_t n_T \sigma_0^2$ decreases with α_{LC} before it becomes constant at a critical value $\alpha_{LC}^* \approx 10^{-5}$. The results for EC follow very similar trends. $L_{opt,EC}(0)/n_t n_T \sigma_0^2$ decreases as α_{EC} is increased until a critical value α_{EC}^* . The final numerical values of LC and EC are in exact agreement for $\alpha_{LC} \leq \alpha_{LC}^*$ and $\alpha_{EC} \geq \alpha_{EC}^*$. This is a strong indicator that the SA algorithm successfully finds the global minimum of Equation 6.2 for the given initial configuration. The difference in efficiency between the two cooling schedules is substantial: for $\alpha_{LC} = 10^{-5}$, the LC schedule requires 10^6 temperature cycles, while the EC schedule only requires about 10^5 cycles to give the same value at $\alpha_{EC} = 0.9999$.

6 Polymer-type-conversion-based target composition

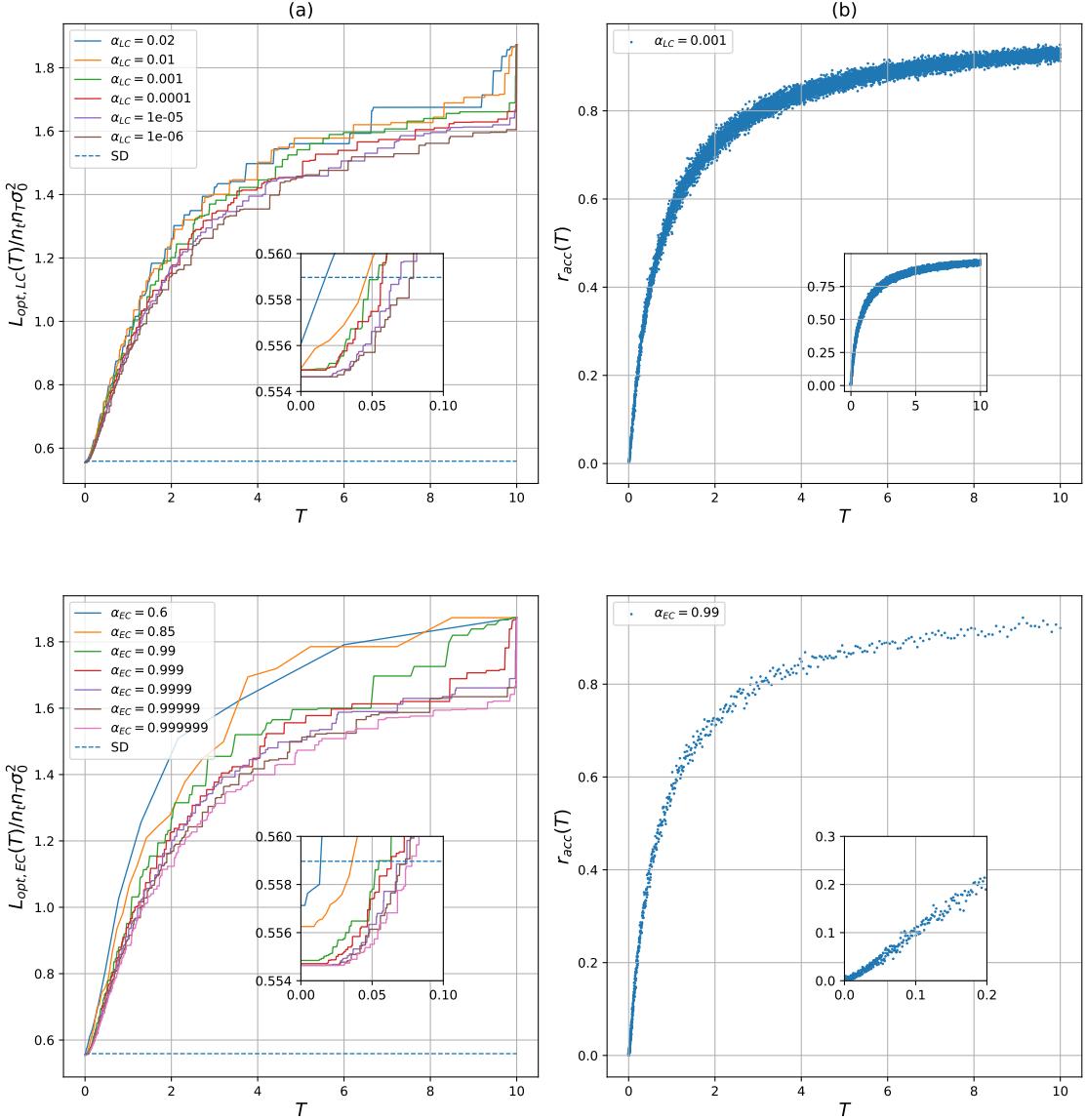


Figure 6.4: (a) Optimum of loss function $L_{opt,i}/n_t n_T \sigma_0^2$ as a function of T for several values of α_i . (b) SA acceptance rate r_{acc} as a function of T . The top shows the results for $i = LC$, the bottom for $i = EC$. The insets show a section that is zoomed in to small temperatures.

The crossover of the SA curve below the SD level happens at a certain temperature T^* , which depends on α_i . This temperature is important since it defines a scale of typical “energy barriers” that the SD algorithm cannot overcome. Figure 6.5 shows the distribution of the change in the loss function Δ for a single flip for different temperatures. For $T = 10$, the distribution is symmetric, which indicates that a flip

may increase or decrease the value of the loss function with the same probability. As T is decreased, less “good” flips are available, so the distribution becomes skewed to the right. For $T = 0.06$, this skew is almost complete.

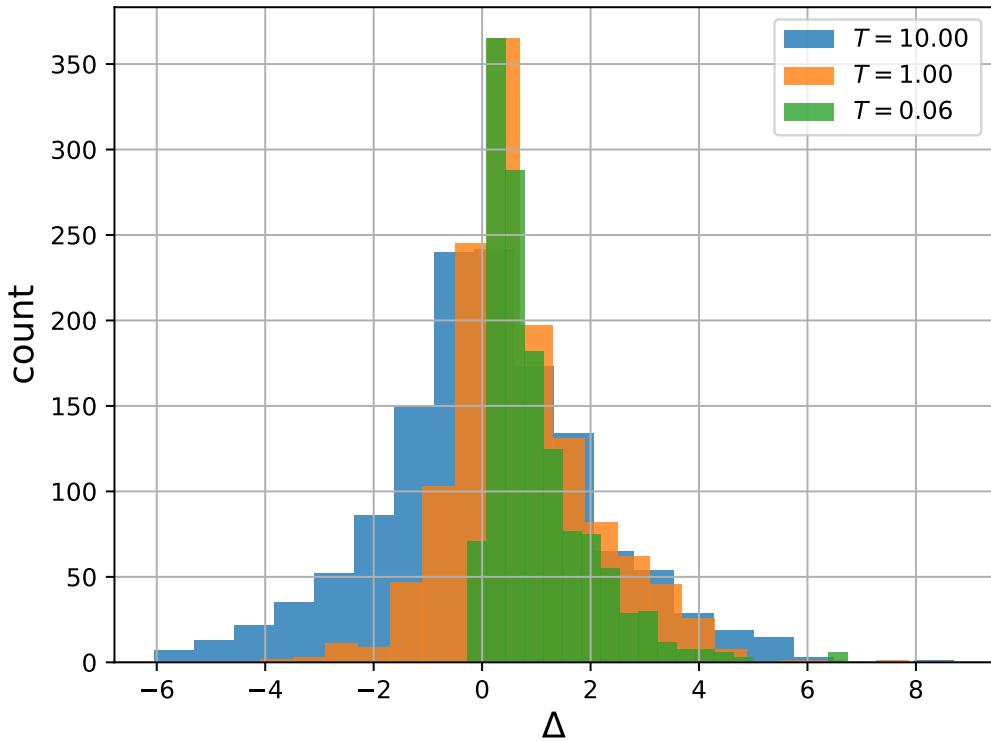


Figure 6.5: Distribution of loss function change Δ per flip attempt at different values of T . EC with $\alpha_{EC} = 0.999$.

The improvement that SA brings compared to SD is very small. Although there is no way to prove that SA finds the global minimum, it is reasonable to assume that the final value is at least close. For the investigated cases, the optimum obtained with SA is less than $0.01\sigma_0^2$ smaller than the SD level. This raises the question of whether SA is even necessary, despite the obvious non-convexity of the problem. The optimized composition profiles $\tilde{\phi}_A(x)$ are shown in Figure 6.6a for $\lambda = 1.25 R_e$ and in Figure 6.6b for $\lambda = 0.25 R_e$. The SA values are obtained from EC with $\alpha_{EC} = 0.999$. The profiles obtained with the different algorithms look very similar. No tendency of the SA profile to match the target profile better than the SD profile can be observed at all. To quantify how different the SA configuration is from the SD configuration, a distance function between two configurations of target polymer types is defined:

$$\text{dist}(\boldsymbol{\beta}, \boldsymbol{\beta}^*) = 1 - \frac{1}{n_p} \sum_{i=1}^{n_p} \delta_{\beta_i, \beta_i^*}. \quad (6.6)$$

One finds $\text{dist}(\boldsymbol{\beta}_{SA}, \boldsymbol{\beta}_{SD}) < 0.05$, so less than 5% of the types are changed in the SA optimum compared to SD.

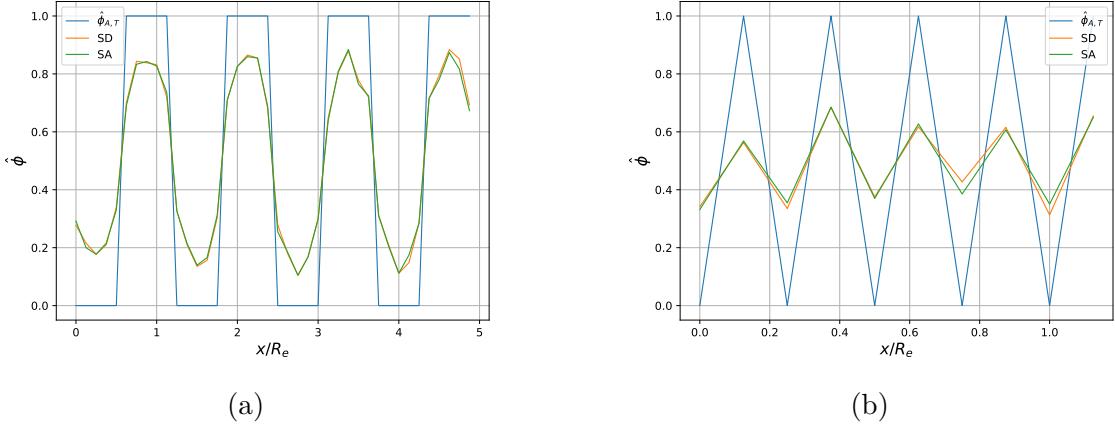


Figure 6.6: Optimized density profile $\hat{\phi}_A(x)$ at $y = 0$, averaged over z . The SA curve is obtained using EC with $\alpha_{EC} = 0.9999$. The target amplitude is $\Delta\tilde{\phi} = 0.5$. (a) $\lambda = 1.25 R_e$. (b) $\lambda = 0.25 R_e$.

6.5 SD simulations

In the previous section, it was concluded that the SA algorithm does not yield significantly better results than the SD algorithm. Additionally, while the number of flip trials performed in the SA algorithm is fixed for a given number of target polymers and temperatures, the SD algorithm converges faster for configurations which were recently optimized than for completely random configurations. To see this, an optimization run is performed on a randomly generated initial configuration and the number of flip proposals $n_{flips,0}$ until convergence is counted. Subsequently, a single MC sweep is performed and the resulting configuration is optimized again, which requires $n_{flips,1}$ flip proposals. The relative difference $\Delta_{flips} = (n_{flips,0} - n_{flips,1})/n_{flips,0}$ is shown in Figure 6.7 as a function of λ , averaged over 100 different initial configurations.

Δ_{flips} increases from about 6% for $\lambda = 0.25 R_e$ to 26% for $\lambda = 5 R_e$. Due to this gain in efficiency, and the previously mentioned reasons, the SD algorithm is chosen

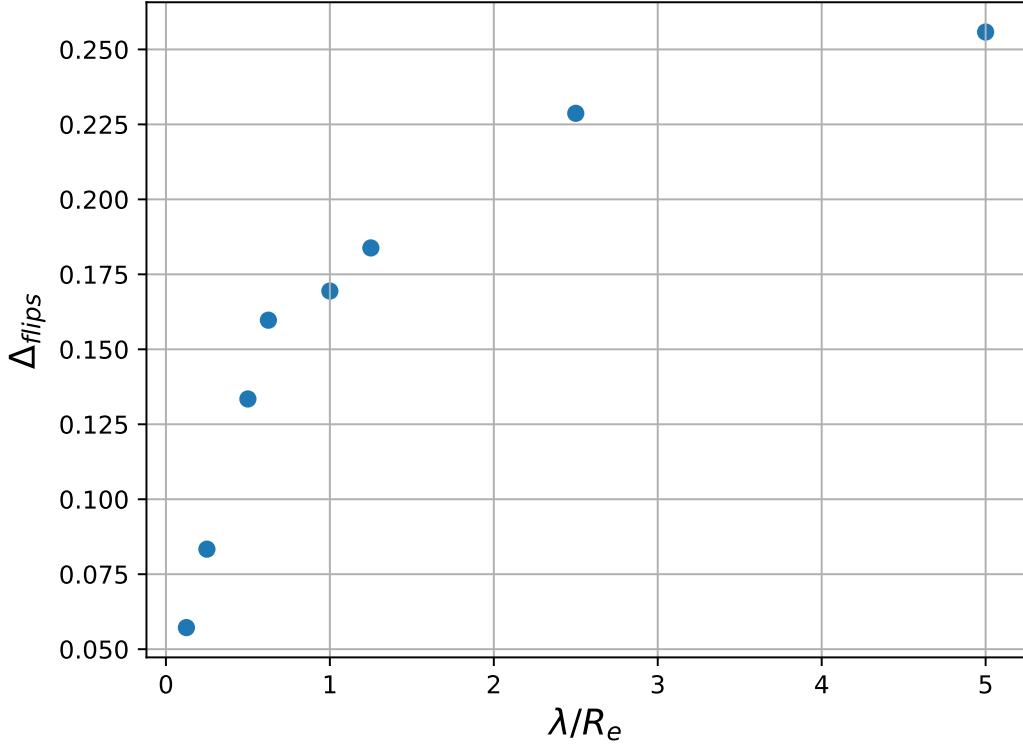


Figure 6.7

for all the following simulations.

One limiting factor in achieving the desired target composition is the lamella periodicity λ . The optimum of the loss function is shown in Figure 6.8b as a function of the lamella wave vector $q = 2\pi/\lambda$ for $\Delta\tilde{\phi} = 0.5$. The values of λ are chosen such that an integer number of lamella periods fit in the simulation box. For large periodicities, the deviation of the composition from the target profile is small everywhere except in the interfacial region between the *A*-rich and the *B*-rich part. The value of the loss function then depends mainly on the number of interfaces, so one finds $L_{opt} \propto q$ for small wave vectors $qR_e/2\pi < 1$. For larger q , or smaller λ , finding the proper type configurations within a lamella becomes more difficult as sharp density changes are required on smaller length scales. This results in a smaller amplitude of the composition, as shown in Figure 6.8a for several values of λ . For $\lambda = 0.5 R_e$, merely an amplitude of about 0.17 is reached. This decrease of the amplitude explains the deviation from the linear behavior in Figure 6.8b for large q .

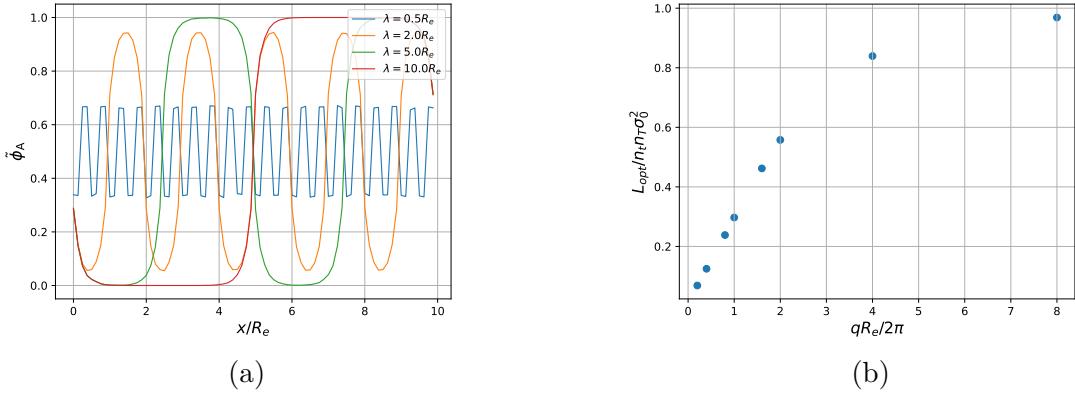


Figure 6.8: (a) One dimensional density profile $\hat{\phi}_A(x)$ at $y = 0$, averaged over z . (b) Normalized loss function optimum as a function of the lamella periodicity λ .

The large deviations of the composition profiles from the target composition, especially for small λ , is not surprising, since the lamellar assembly of noninteracting homopolymers with sharp boundaries is entropically highly disfavored and the chain configurations are thus poorly suited for such target profiles. For polymers with $\chi N \neq 0$, on the other hand, dictating sharp boundaries on small length scales might be desirable. For this, it is necessary to also allow the particle positions to relax.

6.5.1 Impact on bulk

Another issue caused by the finite chain-extension is the extent to which the density is influenced outside the target cells, this can already be seen in Figure 6.3. The composition profile is shown in Figure 6.9a for several values of y . Figure 6.9b shows the maximum composition $\tilde{\phi}_{A,\max}$ as a function of y . It reaches the homogeneous value of 0.5 at a distance of about $1R_e$ from the conversion zone, as one would expect.

6.6 Hybrid strategy

The approach to impose a target composition based only on the type-switching of polymers is flawed and the results are not satisfactory. However, it brings the unique advantage of making the system grand canonical, which is required to simulate the

6 Polymer-type-conversion-based target composition

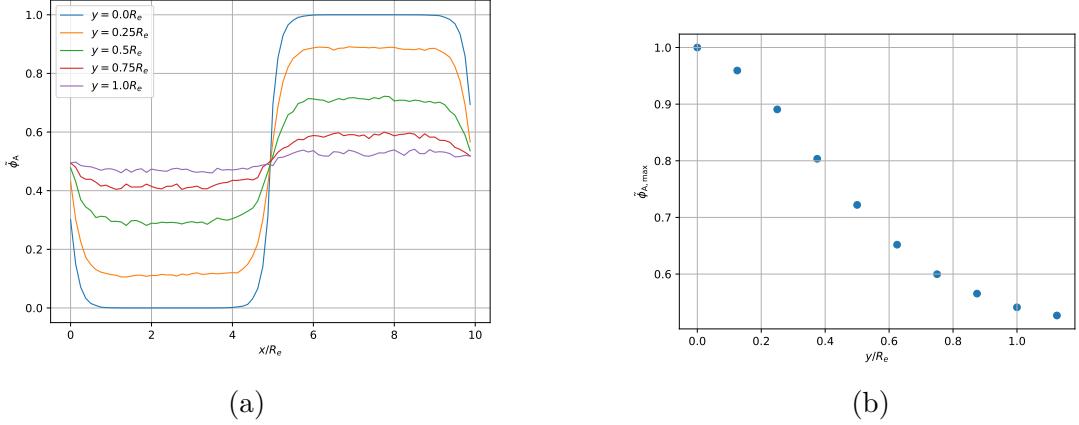


Figure 6.9: (a) 1D density profile of type A for various values of y . (b) Density amplitude as a function of y .

setting described in Figure 1.1. We, therefore, want to modify this approach to better match the target composition. One way to do this would be the use of external fields, as in the previous chapter. In a more physical approach, the interaction part of the quasi-instantaneous fields Equation 2.22 is modified for the target cells:

$$\omega_{A,T}^{\text{inter}}(c) = -\frac{\chi N}{2} (\tilde{\phi}_{A,T}(c) - \tilde{\phi}_{B,T}(c)) , \quad (6.7)$$

and similarly for $\omega_{B,T}^{\text{inter}}(c)$. The new term involves the interaction between the target fields instead of the actual fields. The hybrid scheme now consists of an optimization run using the SD algorithm, followed by a fixed number ΔMCS of simulation steps employing the modified fields Equation 6.7. For $\chi N \neq 0$, this allows the chains to relax after the conversions have taken place.

7 Summary

7.1 Collective diffusion

The key result is that the collective dynamics of homopolymers in a quasi grand-canonical ensemble are accurately described by the simple Onsager coefficient given by Equation 2.17. This leads to a standard diffusion equation Equation 4.4, which involves only the density of species A and the single chain diffusion coefficient D . This essentially means that the system can be considered an ideal gas of polymer chains. While this is partially an artifact of the similarities between the Flory-Huggins theory and regular solution theory, it is surprising that the assumption of local-coupling of the Onsager coefficient, which completely ignores the chain connectivity, gives the correct results.

7.2 External field driven flow

A simple way to implement time-dependent boundary conditions using external fields is presented in this thesis. This method is leveraged to drag lamellar layers of diblock-copolymers through the simulation box, causing them to deform under the experienced friction force. A Péclet number P_e is defined, which describes the ratio between convective and diffusive motion. At a critical number of $P_e \approx 0.85$, the lamellae break. Furthermore, the description of the system is completely analogous to a solid beam bending under a uniform load in the Euler-Bernoulli theory. This is exploited to obtain the bending modulus K of the lamellae. It is found that K increases as N is increased, which would mean that, for a finer chain discretization, more energy is required per unit length to deform the lamellae. This is highly unexpected and may result from a scaling error of the diffusion constant in the SCMF simulations. For $N \rightarrow \infty$, one would expect the value to approach the SCFT value. However, $K_{\text{SCFT}} = 0.16 \sqrt{N} k_B T / R_e$ is significantly smaller than $K_\infty = 0.22 \sqrt{N} k_B T / R_e$.

7.3 Polymer-type-conversion-based target composition

It is shown that the problem of applying non-periodic boundary conditions based on polymer type conversions in SOMA can be interpreted as an optimization problem for the polymer type configurations. The loss function that needs to be minimized is given by Equation 6.2. Two optimization algorithms are presented: the SA algorithm and the SD algorithm. Their performance is compared by taking different test systems and imposing lamellar target compositions at one of the boundaries with amplitude $\Delta\hat{\phi}$ and periodicity λ . While the minimum obtained with SA is always slightly smaller than the one obtained with SD, this difference does not show in the density profiles. Intuitively, it is expected that the SA algorithm performs particularly well for “difficult” target profiles, e.g., large values of $\Delta\hat{\phi}$ and small values of λ . While the latter is indeed observed, the improvement of the SA algorithm actually decreases as a function of $\Delta\hat{\phi}$. It should, however, be emphasized that, for all of the investigated cases, the difference ΔL_{opt} is too small for the SA algorithm to actually produce a density profile that is visibly closer to the target profile than the SD profile. It is therefore one of the key results of this thesis, that the SD algorithm should be used as it yields virtually the same results and is computationally less expensive.

8 Outlook

- Still the old version from Einfuehrung ins Wissenschaftliche Arbeiten

The collective diffusion coefficient of a symmetric homopolymer mixture under a boundary-driven diffusion flux has been studied. In the future, more sophisticated and time-dependent boundary conditions will be employed to manipulate the behavior of the bulk, with the ultimate goal of simulating smaller sections of a large continuum simulation using particle-based simulations. The key question is whether or not the time evolution of the boundary densities is sufficient to dictate the time-evolution of the densities in the bulk. An approach based purely on center-of-mass-based polymer conversion zones might be problematic due to the extension of the converted chains beyond the boundaries. Simple conversions without changing the position of the beads may also lead to unlikely configurations of the affected polymer, so it might be necessary to let the chain grow into the simulation box, e.g. using the configurational bias method [39].

Bibliography

- [1] Ludwik Leibler. Theory of microphase separation in block copolymers. *Macromolecules*, 13(6):1602–1617, 1980.
- [2] Shuanhu Qi and Friederike Schmid. Hybrid particle-continuum simulations coupling brownian dynamics and local dynamic density functional theory. *Soft Matter*, 13:7938–7947, 2017. doi: 10.1039/C7SM01749A. URL <http://dx.doi.org/10.1039/C7SM01749A>.
- [3] Oliver Dreyer, Gregor Ibbeken, Ludwig Schneider, Niklas Blagojevic, Maryam Radjabian, Volker Abetz, and Marcus Müller. Simulation of solvent evaporation from a diblock copolymer film: Orientation of the cylindrical mesophase. *Macromolecules*, 55(17):7564–7582, 2022. doi: 10.1021/acs.macromol.2c00612. URL <https://doi.org/10.1021/acs.macromol.2c00612>.
- [4] Glenn M. Torrie and John P. Valleau. Monte carlo free energy estimates using non-boltzmann sampling: Application to the sub-critical lennard-jones fluid. *Chemical Physics Letters*, 28(4):578–581, 1974. ISSN 0009-2614. doi: [https://doi.org/10.1016/0009-2614\(74\)80109-0](https://doi.org/10.1016/0009-2614(74)80109-0). URL <https://www.sciencedirect.com/science/article/pii/0009261474801090>.
- [5] Ludwig Schneider and Marcus Müller. Multi-Architecture Monte-Carlo (MC) Simulation of Soft Coarse-Grained Polymeric Materials: SOft coarse grained Monte-carlo Acceleration (SOMA). *arXiv e-prints*, art. arXiv:1711.03828, November 2017. doi: 10.48550/arXiv.1711.03828.
- [6] Kostas Ch. Daoulas and Marcus Müller. Single chain in mean field simulations: Quasi-instantaneous field approximation and quantitative comparison with monte carlo simulations. *The Journal of Chemical Physics*, 125(18):184904, 2006. doi: 10.1063/1.2364506. URL <https://doi.org/10.1063/1.2364506>.

Bibliography

- [7] Masao Doi and Sam F Edwards. *The theory of polymer dynamics*. International series of monographs on physics. Oxford Univ. Press, Oxford, 1986. URL <https://cds.cern.ch/record/346518>.
- [8] R.H. Colby M. Rubinstein. *Polymer Physics*. Oxford University Press, 2003.
- [9] Prince E. Rouse. A theory of the linear viscoelastic properties of dilute solutions of coiling polymers. *The Journal of Chemical Physics*, 21(7):1272–1280, 1953. doi: 10.1063/1.1699180. URL <https://doi.org/10.1063/1.1699180>.
- [10] A. Einstein. Über die von der molekularkinetischen theorie der wärme geforderte bewegung von in ruhenden flüssigkeiten suspendierten teilchen. *Annalen der Physik*, 322(8):549–560, 1905. doi: <https://doi.org/10.1002/andp.19053220806>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19053220806>.
- [11] Paul J. Flory. Thermodynamics of high polymer solutions. *The Journal of Chemical Physics*, 10(1):51–61, 1942. doi: 10.1063/1.1723621. URL <https://doi.org/10.1063/1.1723621>.
- [12] P. G. de Gennes. Dynamics of fluctuations and spinodal decomposition in polymer blends. *The Journal of Chemical Physics*, 72(9):4756–4763, 1980. doi: 10.1063/1.439809. URL <https://doi.org/10.1063/1.439809>.
- [13] Ellen Reister. *Zusammenhang zwischen der Einzelkettendynamik und der Dynamik von Konzentrationsfluktuationen in mehrkomponentigen Polymersystemen*. PhD thesis, Mainz, 2002.
- [14] A. N. Semenov. Theory of Long-Range Interactions in Polymer Systems. *J. Phys. II*, 6(12):1759–1780, December 1996. ISSN 1155-4312. doi: 10.1051/jp2:1996159.
- [15] M. W. Matsen and F. S. Bates. Unifying weak- and strong-segregation block copolymer theories. *Macromolecules*, 29(4):1091–1098, 01 1996. doi: 10.1021/ma951138i. URL <https://doi.org/10.1021/ma951138i>.
- [16] Yiyong Mai and Adi Eisenberg. Self-assembly of block copolymers. *Chem. Soc. Rev.*, 41:5969–5985, 2012. doi: 10.1039/C2CS35115C. URL <http://dx.doi.org/10.1039/C2CS35115C>.

Bibliography

- [17] I Ya Erukhimovich and AN Semenov. Nonexponential density relaxation and the dynamic form-factor of polymer melts in the reptation regime. *Zh. Eksp. Teor. Fiz.*, 63:275, 1986.
- [18] H.E Cook. Brownian motion in spinodal decomposition. *Acta Metallurgica*, 18(3):297–306, 1970. ISSN 0001-6160. doi: [https://doi.org/10.1016/0001-6160\(70\)90144-6](https://doi.org/10.1016/0001-6160(70)90144-6). URL <https://www.sciencedirect.com/science/article/pii/0001616070901446>.
- [19] J. G. E. M. Fraaije, B. A. C. van Vlimmeren, N. M. Maurits, M. Postma, O. A. Evers, C. Hoffmann, P. Altevogt, and G. Goldbeck-Wood. The dynamic mean-field density functional method and its application to the mesoscopic dynamics of quenched block copolymer melts. *The Journal of Chemical Physics*, 106(10):4260–4269, 1997. doi: 10.1063/1.473129. URL <https://doi.org/10.1063/1.473129>.
- [20] K. Binder. Collective diffusion, nucleation, and spinodal decomposition in polymer mixtures. *The Journal of Chemical Physics*, 79(12):6387–6409, 1983. doi: 10.1063/1.445747. URL <https://doi.org/10.1063/1.445747>.
- [21] Marcus Müller and Kostas Ch. Daoulas. Speeding up intrinsically slow collective processes in particle simulations by concurrent coupling to a continuum description. *Phys. Rev. Lett.*, 107:227801, Nov 2011. doi: 10.1103/PhysRevLett.107.227801. URL <https://link.aps.org/doi/10.1103/PhysRevLett.107.227801>.
- [22] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 12 2004. ISSN 0021-9606. doi: 10.1063/1.1699114. URL <https://doi.org/10.1063/1.1699114>.
- [23] C. Pangali, M. Rao, and B.J. Berne. On a novel monte carlo scheme for simulating water and aqueous solutions. *Chemical Physics Letters*, 55(3):413–417, 1978. ISSN 0009-2614. doi: [https://doi.org/10.1016/0009-2614\(78\)84003-2](https://doi.org/10.1016/0009-2614(78)84003-2). URL <https://www.sciencedirect.com/science/article/pii/0009261478840032>.

Bibliography

- [24] Marcus Müller and Kostas Ch. Daoulas. Single-chain dynamics in a homogeneous melt and a lamellar microphase: A comparison between Smart Monte Carlo dynamics, slithering-snake dynamics, and slip-link dynamics. *The Journal of Chemical Physics*, 129(16), 10 2008. ISSN 0021-9606. doi: 10.1063/1.2997345. URL <https://doi.org/10.1063/1.2997345>. 164906.
- [25] Michael R Gary and David S Johnson. Computers and intractability: A guide to the theory of np-completeness, 1979.
- [26] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, 2nd edition, 2001. ISBN 0262032937. URL <http://www.amazon.com/Introduction-Algorithms-Thomas-H-Cormen/dp/0262032937%3FSubscriptionId%3D13CT5CVB80YFWJEPWS02%26tag%3Dws%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3D0262032937>.
- [27] Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983. doi: 10.1126/science.220.4598.671. URL <https://www.science.org/doi/abs/10.1126/science.220.4598.671>.
- [28] P. J. M. Laarhoven and E. H. L. Aarts. *Simulated Annealing: Theory and Applications*. Kluwer Academic Publishers, USA, 1987. ISBN 9027725136.
- [29] Bruce Hajek. Cooling schedules for optimal annealing. *Math. Oper. Res.*, 13(2):311–329, may 1988. ISSN 0364-765X.
- [30] J. Baschnagel, H. Meyer, F. Varnik, S. Metzger, M. Aichele, M. Müller, and K. Binder. Computer simulations of polymers close to solid interfaces: Some selected topics. *Interface Science*, 11(2):159–173, 2003. doi: 10.1023/A:1022118610890. URL <https://doi.org/10.1023/A:1022118610890>.
- [31] Frank S. Bates, Wayne W. Maurer, Paul M. Lipic, Marc A. Hillmyer, Kristoffer Almdal, Kell Mortensen, Glenn H. Fredrickson, and Timothy P. Lodge. Polymeric bicontinuous microemulsions. *Phys. Rev. Lett.*, 79:849–852, Aug 1997. doi: 10.1103/PhysRevLett.79.849. URL <https://link.aps.org/doi/10.1103/PhysRevLett.79.849>.

Bibliography

- [32] M. Müller and G. Gompper. Elastic properties of polymer interfaces: Aggregation of pure diblock, mixed diblock, and triblock copolymers. *Phys. Rev. E*, 66:041805, Oct 2002. doi: 10.1103/PhysRevE.66.041805. URL <https://link.aps.org/doi/10.1103/PhysRevE.66.041805>.
- [33] Dominik Dürchs, Venkat Ganesan, Glenn H. Fredrickson, and Friederike Schmid. Fluctuation effects in ternary ab + a + b polymeric emulsions. *Macromolecules*, 36(24):9237–9248, 12 2003. doi: 10.1021/ma030201y. URL <https://doi.org/10.1021/ma030201y>.
- [34] Marcus Müller. Studying amphiphilic self-assembly with soft coarse-grained models. *Journal of Statistical Physics*, 145:967–1016, 11 2011. doi: 10.1007/s10955-011-0302-z.
- [35] C. Pangali, M. Rao, and B.J. Berne. On a novel monte carlo scheme for simulating water and aqueous solutions. *Chemical Physics Letters*, 55(3):413–417, 1978. ISSN 0009-2614. doi: [https://doi.org/10.1016/0009-2614\(78\)84003-2](https://doi.org/10.1016/0009-2614(78)84003-2). URL <https://www.sciencedirect.com/science/article/pii/0009261478840032>.
- [36] Peter J. Rossky, Jimmie D. Doll, and Harold L. Friedman. Brownian dynamics as smart monte carlo simulation. *Journal of Chemical Physics*, 69:4628–4633, 1978.
- [37] Bastian E. Rapp. Chapter 9 - fluids. In Bastian E. Rapp, editor, *Microfluidics: Modelling, Mechanics and Mathematics*, Micro and Nano Technologies, pages 243–263. Elsevier, Oxford, 2017. ISBN 978-1-4557-3141-1. doi: <https://doi.org/10.1016/B978-1-4557-3141-1.50009-5>. URL <https://www.sciencedirect.com/science/article/pii/B9781455731411500095>.
- [38] Zhen-Gang Wang. Response and instabilities of the lamellar phase of diblock copolymers under uniaxial stress. *The Journal of Chemical Physics*, 100(3):2298–2309, 02 1994. ISSN 0021-9606. doi: 10.1063/1.466528. URL <https://doi.org/10.1063/1.466528>.
- [39] Jörn Ilja Siepmann and Daan Frenkel. Configurational bias monte carlo: a new sampling scheme for flexible chains. *Molecular Physics*, 75(1):59–70, 1992. doi: 10.1080/00268979200100061. URL <https://doi.org/10.1080/00268979200100061>.