View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Utrecht University Repository

Simulation of Polymer Dynamics in Gels and Melts

Simulation of Polymer Dynamics in Gels and Melts

Simulatie van Polymeerdynamica in een Gel of Smelt
(met een samenvatting in het Nederlands)

Proefschrift

ter verkrijging van de graad van doctor aan de Universiteit Utrecht op gezag van de Rector Magnificus, Prof. dr. W. H. Gispen, ingevolge het besluit van het College voor Promoties in het openbaar te verdedigen op maandag 1 december 2003 des middags te 12.45 uur

door

Alexander van Heukelum

geboren op 9 mei 1975, te Rotterdam

promotor: Prof. dr. H. van Beijeren

copromotor: Dr. G. T. Barkema

Faculteit Natuur- en Sterrenkunde
Universiteit Utrecht

ISBN 90-393-3574-5

Contents

Table of Contents **i**

Preface **iii**

1 Introduction **1**

1.1 Polymers 1
1.2 Polymer solutions 2
1.3 External fields 2
1.4 Polymers in a gel 3
1.5 Gel electrophoresis 5
1.6 Polymers and polymeric mixtures in a melt or dense solution . . 8

2 Lattice models for polymeric systems **9**

2.1 Self-avoiding random walks 9
2.2 Bond-fluctuation model 10
2.3 Cage model 11
2.4 Repton model 13
2.5 Comparison of the models 14

3 Gel electrophoresis **19**

3.1 Bond-fluctuation model 20
3.2 Cage model 20
3.3 Repton model 21
3.4 Differences between the cage and repton models 22

4 Electrophoresis simulated with the cage model **25**

4.1 Multispin coding 25
4.2 Drift velocity 28
4.3 Polymer shapes 29

5 Exact enumeration of the cage model for electrophoresis 35
5.1 State space and the transition matrix 35
5.2 Power method 37
5.3 Reducing the number of states 37
5.4 Reduced transition matrix 40
5.5 Results 41

6 Extensions of the repton model for simulating melts 45
6.1 Repton model for polymer melts 45
6.2 Implementation of the projected repton model 47
6.3 Implementation of the extended repton model 50
6.4 Phase separation of a binary polymer mixture 52

7 Fractionation 57
7.1 Experimental research 57
7.2 Flory-Huggins theory 58
7.3 Computer simulations 63
7.4 Comparisons 67

A Parallel matrix-vector multiplication 69
A.1 BSP 69
A.2 Matrix distribution 70
A.3 Exploiting the sparsity structure 72
A.4 Timings 74

B Proof of correctness of the kink-representation approach 77

Bibliography 79

Samenvatting (summary in Dutch) 87

Dankwoord (acknowledgements) 93

Curriculum Vitae 95

Preface

This thesis contains the work I did during my work as O.I.O. at the Institute for Theoretical Physics at Utrecht university. The work can roughly be divided into two parts. The first part of the research is on the motion of a lattice polymer in a gel, subject to a force that drags the polymer in one direction. This is a good model for DNA electrophoresis which is an experimentally well-studied subject. A few lattice polymer models for electrophoresis have been investigated in literature [1] (see chapters 2 and 3). We have computed drift velocities in the cage model of electrophoresis for long polymers with Monte Carlo simulations [2] (see chapter 4). For polymers up to a length of $L = 15$, we were able to compute numerically exact the probability of any configuration of a polymer in an applied field [3] (see chapter 5). The second part describes my work on binary polymer mixtures in the framework of a lattice model. This part can be divided into two subjects: the equilibrium state and the dynamics of phase separation. These two subjects have been studied experimentally [4, 5], parallel to the theoretical and computational work I have performed. The rate of demixing of a phase-separating polymer mixture may be characterized by the growth properties of the domains [6] (see chapter 6). In chapter 7 we investigate the composition of the two equilibrium states resulting after full phase separation [7].

Chapter 1

Introduction

This chapter introduces important physical properties of the polymeric systems considered in this thesis, appropriate approximations to describe the long-time and equilibrium physics, and analytic theories that have been used to describe the physical behaviour of such systems in earlier scientific publications.

1.1 Polymers

The characteristic property of a polymer is that it consists of a large number of repeating units. Often, polymers are manufactured by a concatenation reaction of monomers. The repeating unit is therefore often referred to as “monomer”. Polymers, and more specifically plastics, form an interesting class of materials that is very important to industry, because different types of polymers show a wide range of desirable characteristics for various large-scale applications. An example of such a polymer widely used in industry is polyethylene which consists of chemically connected ethylene molecules. Each monomer can connect chemically with two other monomers. When two monomers react, a dimer comes into existence, which itself can connect with two other monomers or dimers. Monomers, dimers, trimers, and polymers can connect indefinitely in this way, and long linear chains come into existence.

Nature also has created several polymers. These polymers are used in the cell to provide strength, like keratin and actin; or to encode genetic information, like DNA and RNA. Other natural polymers include proteins, but natural proteins bond strongly to themselves forming globules which are often more appropriately modeled as spheres than as chains. The natural polymers often have a number of different repeating units that are connected in a semi-random way. For example, the ordering of four types of amino acids in DNA encodes the genetic information of a cell.

Certain natural polymers are used in the food industry. The two types of polymers used in the experimental work related to our investigations [4,5,8] are dextran and gelatin. Dextran is a well-defined polymer with repeating glucose units, and gelatin is a natural polymer that consists of a semi-random chain of amino acids.

1.2 Polymer solutions

A polymer in a solvent is usually assumed to be in a regime where the instantaneous velocities of segments of the chain are damped very quickly by the viscosity of the surrounding solvent. Only quasi-instantaneous fluctuations in the positions of segments of the polymer play an important role in the mobility. The rates of diffusion of segments along the polymer chain and perpendicular to the chain may be different, but the diffusion of the polymer as a whole is well described by Brownian dynamics. The diffusion coefficient may be measured as the ensemble average of the square of the distance travelled by a polymer per unit of time, i.e.,

$$D = \lim_{t \to \infty} \langle |\vec{r} - \vec{r_0}|^2 \rangle / 2d(t - t_0), \tag{1.1}$$

where $\vec{r}$ and $\vec{r_0}$ are the positions of a reference point of a d-dimensional polymer at times t and t_0 respectively. In the regime where there is no entanglement between the polymers, nor entanglement between the polymer and its environment, and no hydrodynamic interaction either—this is the the so-called Rouse regime—the diffusion coefficient D is inversely proportional to the polymer length L:

$$D \sim 1/L. \tag{1.2}$$

1.3 External fields

If a small force is applied to the polymer, and the dynamics of the segments in absence of the applied force is well described by Brownian dynamics, then the fluctuation-dissipation theorem (1.3), also known as the Nernst-Einstein relation, applies. It states that the drift velocity $\vec{v}$ induced by a driving force $\vec{F}$ is linearly proportional to the driving force. The proportionality constant equals, up to a factor of $k_B T$, the diffusion coefficient D in the absence of external forces:

$$\vec{v} = (D/k_B T)\, \vec{F}, \tag{1.3}$$

with Boltzmann constant k_B and absolute temperature T.

This can be applied to polyelectrolytes like DNA. Since DNA is acidic, it becomes negatively charged when it is dissolved in water. For our purpose,

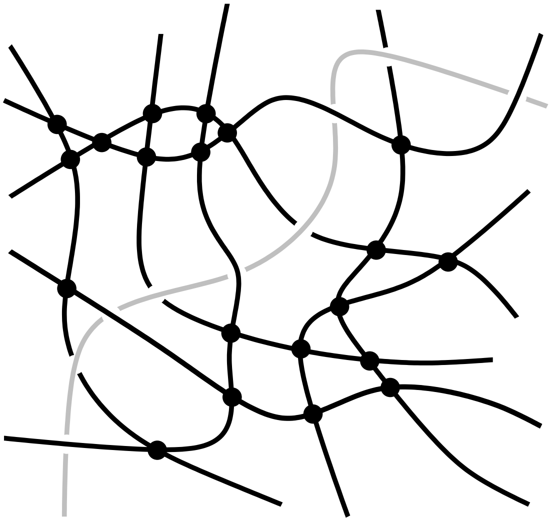

Figure 1.1: Impression of a single polymer (gray) in an environment of cross-linked gel strands.

water-dissolved DNA can be treated as a polyelectrolyte containing L monomers, where each monomer has a fixed electric charge q. If an electric field is applied, this causes the DNA to migrate in one direction. The driving force $\vec{F}$ resulting from an electric field $\vec{\mathcal{E}}$ is proportional to the chain length:

$$\vec{F} = qL\vec{\mathcal{E}}. \tag{1.4}$$

Combining equation (1.3) with equations (1.2) and (1.4), one finds that the velocity depends on q and $\vec{\mathcal{E}}$, but not on the polymer length L. Thus, in the regime of Rouse dynamics, a polydisperse mixture of polyelectrolytes will not separate under the influence of a constant electric field.

1.4 Polymers in a gel

The situation becomes quite different (and theoretically more challenging) for a polymer solution in a gel. A *gel* consists of polymers or *gel strands* that are cross-linked, forming a stable three-dimensional network, like in figure 1.1. The *pores*, spaces between the gel strands, are filled with a solvent. The gel strands impose an important restriction onto the motion of the polymers: when a polymer moves perpendicular to its length axis, the movement will soon be blocked by the gel strands. The gel effectively does not allow sideways movement

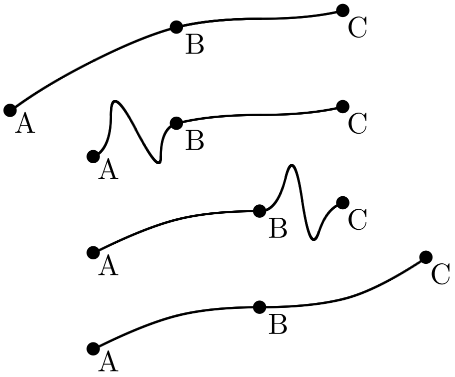

Figure 1.2: Reptation of a polymer. If the sideways motion of a polymer is highly suppressed, motion of stored length along the backbone of the polymer becomes the most important contribution to the diffusion. One of the ends of the polymer seeks the way and all other monomers must follow the same path. To model this longitudinal motion, certain defects are allowed to move along the polymer. A defect can come into existence at either end of the chain, contracting the polymer by a certain amount of length, called the stored length of the defect. The defect can now travel along the polymer, and when it reaches the other end of the chain it disappears, releasing its stored length by extending the chain on that side.

of the polymer outside a certain *tube*, confined by the gel strands surrounding the polymer.

Experiments on polymers in a gel have often been performed for DNA in agarose gels. It is known that below a certain temperature, agarose forms long strands which cross-link and impede movement of the polymer transverse to its length. In 1971, De Gennes [9] described the dynamics of a polymer in such a cross-linked gel. The way to describe long-time diffusion was to introduce the notion of *defects* along the chain, which represent *stored length.* These defects diffuse along the chain, and they come into existence and annihilate only at the ends of the polymer, as shown in figure 1.2. This type of motion is called *reptation.* From this model De Gennes concluded that, for long polymer chains of length L in a gel:

- the relaxation time set by the diffusion of defects along the polymer chain scales as L^2;
- the typical time needed for the polymer chain to leave its tube scales as L^3;

- the diffusion coefficient D scales as

$$D \sim L^{-2}; \tag{1.5}$$

- these dynamical properties of the polymer are insensitive to the type of defects introduced, as long as those do not allow sideways movements of the tube.

Numerical simulations to check these results of reptation theory at first reported exponents for the L-dependence of the diffusion coefficient that differed from De Gennes' prediction, equation (1.5) [10], but with the advance of computational power as well as computational techniques the numerical evidence supporting De Gennes' scaling prediction has become compelling [11, 12].

1.5 Gel electrophoresis

In the rapidly-growing fields of molecular genetics and genetic engineering, gel electrophoresis is a technique of great importance. One reason is that it enables efficient separation of polymer strands by length. In DNA electrophoresis, strands of DNA of various lengths are injected into a gel composed of agarose and a buffer solution. In solution, DNA becomes negatively charged with a fixed charge per unit length. When an electric field is applied, it provides a driving force for the DNA, and the DNA moves towards the positive electrode.

Like any other polymer, DNA consists of a large number of connected monomers. A DNA monomer consists of a *base pair* which contributes about 2.5Å to the length of the DNA strand. The microscopic structure is the well-known double helix structure of DNA in three dimensions, which makes the DNA polymer quite rigid. The diameter of the double helix is about 20Å and the distance between two turns is about 35Å. The *persistence length* defines the typical length over which the polymer preserves its orientation. For DNA it is much larger than the diameter of the double helix structure; one persistence length usually contains between 130 and 375 base pairs depending, for instance, on temperature and salinity [13]. It is well known that DNA contains the genetic code of living beings. Each cell in one organism contains the same genetic information, since it is just duplicated at each cell division. Retrieving the DNA from cells is a relatively easy task, and splitting the DNA at certain places, using enzymes, has become a standard procedure. The resulting mixture of DNA fragments is unique for each individual, like a fingerprint. When the DNA fragments are separated by length by means of gel electrophoresis, this unique signature is made visible. Also, genetic mutations from one generation to the next can be made visible this way.

In spite of the frequent use of electrophoresis in genetics and biochemistry, the dynamics of polymers such as DNA in a gel are not completely understood. Given the great practical importance of DNA electrophoresis, there is much interest in gaining an understanding of precisely what the mechanisms of gel electrophoresis are and how the migration rate depends on length, applied electric field, and the properties of the agarose gel. Experimentally it is found that, as long as the force on each DNA fragment is below a certain threshold, the drift velocity of the fragments is inversely proportional to the length of the fragments and directly proportional to the applied electric field, consistent with the combination of equations (1.3), (1.4), and (1.5):

$$v \sim (L\mathcal{E}) \cdot L^{-2} = \mathcal{E}/L. \tag{1.6}$$

Since the drift velocity depends on the length of the fragments, after some time the mixture of DNA fragments separates into a number of bands, each consisting of DNA fragments with the same length and thus the same velocity. Shorter fragments are located in bands that have moved further from the point of injection. This technique allows the isolation of fragments with a particular length. If the applied electric force on the DNA fragments is above a certain threshold, the velocity becomes independent of the length of the fragment: the bands collapse. The longer the fragments, the lower this treshold becomes. This makes the separation of long DNA fragments with electrophoresis a difficult task.

Theoretically, the motion of a reptating polymer in a field can be explained by a bias of the direction of end segments as they extend the tube [14–16]. On average, these end segments are oriented in the direction of the applied force. Once the entire chain has left its original tube, the directional bias of the end segment has been passed on to the remainder of the chain, which simply followed the end. Thus, in time, the whole chain receives this directional bias. All reptation models show drift velocities linear in the applied field, and inversely proportional to the polymer length. This result holds at least as long as the polymer configurations are not strongly disturbed by the presence of the electric field, which is true at least in the limit of small strengths of the electric field, $\mathcal{E} < c/L$.

Once the force $qL\mathcal{E}$ on the polymer exceeds a certain threshold value the Nernst-Einstein relation fails. Experimentally, it is known that for a field strength that is sufficiently strong so that the fluctuation-dissipation theorem does not apply any longer, the velocity increases superlinearly with the applied field, and becomes length-independent—so-called *plateau mobility*. It is clear that the velocity of a polymer of length L in an electric field with strength $\mathcal{E}$ should be an odd function of $\mathcal{E}$: if the electric field is reversed, the polymers move in the opposite direction with the same velocity. Thus, a series expansion of the drift velocity as a function of electric field strength should contain only odd powers of the electric field strength:

$$v(L, \mathcal{E}) = \sum_{i \text{ odd}} a_i(L)\, \mathcal{E}^i. \tag{1.7}$$

Indeed, the analytically known results for short polymers in one of the models of reptation (the repton model as discussed in section 3.3) can be written in this form [17]. Based on these arguments, as well as on scaling arguments within the biased reptation model, the incorrect conclusion was drawn that thus in the plateau mobility regime $v \sim \mathcal{E} + \mathcal{O}(\mathcal{E}^3)$; expressed in the mobility $\mu = v/\mathcal{E}$, this is identical to $\mu \sim 1 + \mathcal{O}(\mathcal{E}^2)$.

In the beginning of the nineties, Duke, Semenov, and Viovy [18] showed that fluctuations in the stored length along the chain radically alter the field-dependence of the tube orientation. Their analysis suggests a different scaling as

$$v \sim \mathcal{E} + \mathcal{O}(\mathcal{E}^2), \tag{1.8}$$

at odds with earlier arguments that supported $v \sim \mathcal{E} + \mathcal{O}(\mathcal{E}^3)$.

Experiments support this $\mathcal{O}(\mathcal{E}^2)$ scaling of the drift velocity in the region of plateau mobility [19], and this scaling was already clearly evident in earlier experimental results (but not noticed), for instance in those obtained by Hervet and Bean [20], figure 2. The reconciliation with the previous line of arguments lies in the observation that the series expansion, equation (1.7), converges only if $L\mathcal{E}$ is small enough.

An intuitive explanation of the dependence of the drift velocity on the electric field strength goes as follows [21]: continuous chains and Monte Carlo chains transmit tension by an entropic process. A random polymer will have an end-to-end length around $h = c\sqrt{L}$. When an electric field is applied, the polymer is stretched in the direction of the electric field. A stretched polymer configuration is entropically less favorable than a compact form: the result is an elastic force that contracts the polymer. When the electric field exceeds a certain level, the polymer as a whole does no longer resemble a random walk ($h > c\sqrt{L}$). One may cut the polymer into n_b pieces (*blobs*) of length $L_b = L/n_b$, that each still look like a random walk; the average end-to-end distance of the blobs is equal to $\langle h_b \rangle = c\sqrt{L_b}$. The elastic force is proportional to the size of the blob and inversely proportional to the length of the part of the polymer that forms the blob: $F_{\text{elastic}} \sim h_b/L_b$. The electric force on a blob is proportional to the size of the blob as well as the electric field: $F_{\text{electric}} \sim h_b\mathcal{E}$. These two forces have to be in balance which implies that the blob size is $L_b \sim \mathcal{E}^{-1}$. The Nernst-Einstein relation now applies to the blobs, so $v = F_b D_b = qL_b\mathcal{E}D_b$. If the blob size is large enough, $D_b \sim L_b^{-2}$ which makes the speed of the polymer quadratic in the electric field: $v \sim q\mathcal{E}/L_b \sim q\mathcal{E}^2$. This effect has been observed in the repton model [21], as well as in the cage model [2], which we will discuss in chapters 2 and 3.

In very strong electric fields, polymers tend to align with the direction of the field either completely, in which case they move with a high velocity, or partially, in which case they assume a U-shaped configuration with hardly any mobility. Unless specific long-ranged moves are included, these U-shaped configurations result in an exponentially decreasing drift velocity with increasing field strength [22,23]. Experimentally, the dominance of U-shaped configurations is confirmed [24,25], but the proposed exponential decrease in drift velocity is not.

1.6 Polymers and polymeric mixtures in a melt or dense solution

Polymers in a melt or dense polymer solution are entangled. To a single polymer, the surrounding polymers in the melt or dense solution act like a gel; if the collective motion of polymers is neglected, the description of a polymer in a gel and the description of a polymer in a melt are equal: sideways movement is also largely prohibited in a dense polymer solution or a polymer melt. Therefore, the main mode of movement is believed to be reptation-like. Indeed, Perkins *et al.* [26], showed that a polymer in a melt is confined by a ‘tube’, by dragging a marked DNA strand through a dense solution of DNA strands.

In a solution or melt with two polymeric constituents, another type of dynamical phenomenon can occur. If an effective repulsion between the two types exists, the system will phase-separate below a certain critical temperature. This dynamical phenomenon cannot be described by considering a single polymer, since inter-chain interactions provide the driving force for the phase separation.

If two types of polymer are present in a single solvent, and the polymers show an effective repulsive interaction with each other, then the mixture can still be homogeneously mixed at sufficiently high temperatures. Below a certain temperature, the system separates into two phases. If the temperature is lowered rapidly from a value above the phase-separation temperature to one below it, the polymers will separate locally into small domains. Each domain is rich in one of the polymer types and dilute in the other type. After the initial phase separation, the typical size of the domains will grow steadily.

After full phase separation, the phases reach equilibrium. Entropy considerations show that in general the phases are not pure: a small but significant fraction of each phase consists of the diluted polymer type. If the polymer types are not monodisperse, the molar mass distributions of the same polymer type in the two coexisting phases become different from each other [4,5,7,8]. This is known as *fractionation.*

Chapter 2

Lattice models for polymeric systems

The description of polymers with highly simplified lattice models dates back to at least 1939 [27], and is perhaps even older, although we could not find earlier references. With the arrival of computer simulations, lattice polymer models have gained in significance. Below, we list four of the most common types of lattice models used for the description of polymers.

2.1 Self-avoiding random walks

Already in 1962, the dynamics of polymers has been modeled with self-avoiding walks on square and cubic lattices [28]. The dynamics consists of single-monomer moves. Monomers can move to any unoccupied lattice site, provided the bond lengths to their neighbors along the chain remain one lattice unit. For monomers in the interior of the chain this means that only two adjacent bonds at 90° angles can be exchanged. The timescale is set by attempting to move each monomer on average once per unit of time in each direction. Verdier and Stockmayer [28] computed equilibrium distributions of the end-to-end distance of the polymer chain and relaxation properties of initially fully stretched chains.

The self-avoiding walk model provides a good model only for polymer dynamics in a highly diluted polymer solution. In semi-dilute and dense solutions the polymers are better described by random walk statistics, where the end-to-end distance of a long polymer of length L in a melt scales as L^{ν}, with $\nu = 1/2$, instead of $\nu = 0.588$. This has been shown with self- and mutually avoiding walks on a diamond lattice [29, 30], which gives $\nu = 0.50 \pm 0.01$ instead of $\nu = 0.59 \pm 0.01$, which was found for a single self-avoiding walk on the same lattice. Equivalent results are seen for self- and mutually avoiding random

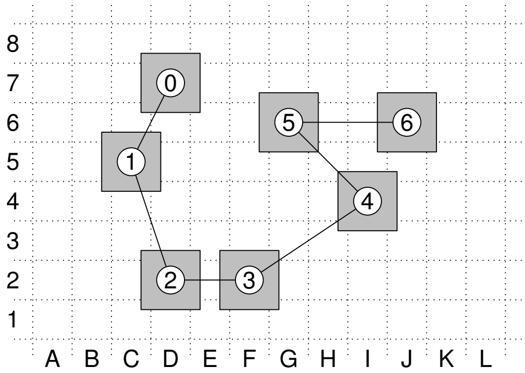

Figure 2.1: The 2D bond-fluctuation model. Monomers are located on a square lattice. Adjacent monomers are connected by a bond, which can only take a length of 2, $\sqrt{5}$, $2\sqrt{2}$, 3, $\sqrt{10}$, or $\sqrt{13}$. The excluded area around each monomer consists of 2×2 lattice sites. The particular choice for the values allowed for the bond length, in combination with the size of the excluded area, suffices to avoid the crossing of bonds. To simulate the effect of the gel, obstacles with a size of 2×2 lattice sites are introduced in the lattice. In the figure one such an obstacle is shown as a dark-gray square. The dynamics of the polymer chain consists of single-monomer hops to nearest-neighbor lattice sites, restricted by the constraints on bond length and excluded volume.

walks up to length 800 on a cubic lattice with local dynamics that allowed up to two adjacent monomers to move at once [31,32]. The diffusion constant for this model was found to scale with length as $D \sim L^{-1.52\pm0.06}$, and the longest relaxation time as $\tau \sim L^{2.63\pm0.04}$.

Recently, Hu *et al.* [33] reported dynamic Monte Carlo simulation results on a model that uses self-avoiding walks on a cubic lattice where bonds can connect sites at distances 1, $\sqrt{2}$, and $\sqrt{3}$.

2.2 Bond-fluctuation model

The two-dimensional bond-fluctuation model, introduced by Carmesin and Kremer in 1988 [34,35], is the most microscopically detailed lattice polymer model described in this chapter. In this model, each monomer occupies four (2×2) lattice sites of a square lattice. Thus, two monomers are always separated by at least a distance of two lattice spacings. Monomers adjacent in the chain are connected by bonds with lengths between 2 and $\sqrt{13}$. The model is sketched in figure 2.1. The dynamics consist of single-monomer moves over a single lattice spacing, i.e., monomers can move up, down, left, or right by one unit of length.

In the figure, for example, monomer 1 cannot move up because of the excluded-volume constraint as well as the bond-length constraint. The time scale is set such that each possible monomer move is attempted statistically once per unit of time.

The model has been extended to three dimensions [36]: each monomer occupies eight ($2 \times 2 \times 2$) sites of a cubic lattice, and the distance between adjacent monomers is constrained to certain values, never exceeding $\sqrt{10}$. The allowed bond lengths are chosen such that it is impossible for a third monomer to slip through between two connected monomers when it is moved by one lattice site at a time. The three-dimensional bond-fluctuation model has been used to describe the behavior of a dense polymer melt. Paul *et al.* [37] performed simulations of 10^7 Monte Carlo steps with chains up to $L = 200$ (about 6 entanglement lengths for 50% filling of the lattice) on a cubic lattice containing 40^3 lattice sites. They report the onset of reptation dynamics, in which the correlation time τ scales as $\tau \sim L^3$. One decade later, simulations of this model with polymers of up to $L = 512$ monomers (i.e., a curvilinear length of about 1350 for a filling fraction of 50% of the lattice, or about 14 entanglement lengths), were performed [38]. It was concluded that the crossover from non-entangled Rouse dynamics to entangled reptation dynamics is "very protracted". The shortest chains did not behave Rouse-like, but the longest chains were not in the asymptotic reptation regime either. Other uses of the bond-fluctuation model include the study of adsorption of polymers at a hard wall [39] and simulations of entangled closed-ring polymers [40].

2.3 Cage model

In a series of three papers in 1981, Evans and Edwards [41–43] used the random-walk equivalent of the model of Verdier and Stockmayer, described in section 2.1, to study the dynamics of polymers in a gel. Their version of the model is generally called the *cage model*, and it is also sometimes referred to as the Evans-Edwards model or De Gennes' model for polymer dynamics. Two adjacent monomers, connected by a "bond", must reside in adjacent sites of a cubic lattice. No excluded-volume interactions are enforced, so each chain may visit the same lattice site repeatedly. This ensures that the end-to-end distance depends on polymer length as L^ν, with $\nu = 1/2$. Figure 2.2 shows an impression of the model. Figure 2.3 shows the two-dimensional version of the cage model.

The dynamics of the cage model consists of a sequence of single-monomer steps: either an exchange of two consecutive steps under an angle of 90°, or a reorientation of a "kink" (two consecutive steps in opposite directions). The bond of an end monomer is randomly replaced by a bond in one of the six possi-

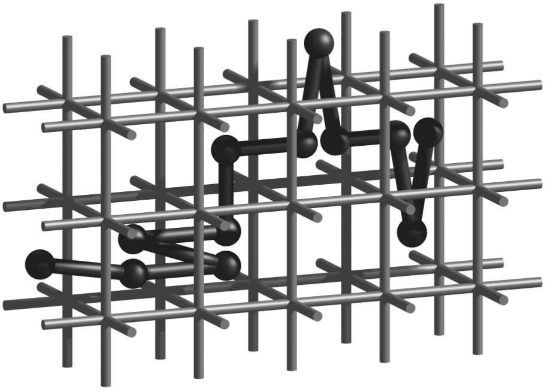

Figure 2.2: The three-dimensional cage model. In the cage model, many monomers can occupy the same lattice site. For clarity of the figure, those monomers are drawn separately, but close to each other.

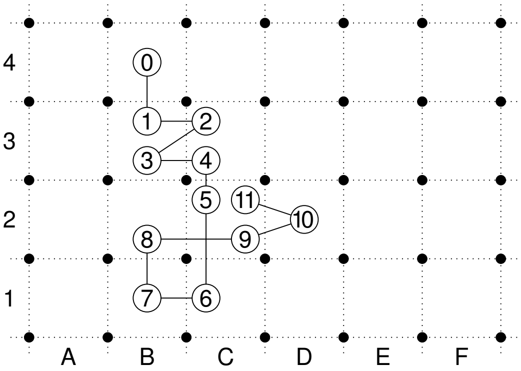

Figure 2.3: The two-dimensional cage model. Monomers are located in the middle of faces of a square lattice (here sometimes slightly displaced for clarity). Monomers adjacent in the chain are connected through bonds with unit length. The dynamics consist of single-monomer steps, constrained to preserve unit-length bonds. In the cage model, a gel is modeled by fixed obstacles located on the lattice sites. The obstacles cannot be crossed by the chain. If all lattice sites are occupied by obstacles (the usual situation, and the one depicted here), the only type of movement available to monomers in the interior of the chain is the reorientation of "kinks" (pairs of adjacent anti-parallel steps, such as those between monomers 2, 3, and 4).

ble directions. The kink moves, 90° moves, and end moves are the only possible moves. The edges of a cubic lattice, translated by a vector $(1/2, 1/2, 1/2)$ relative to the lattice on which the polymer resides, contain the gel strands that form a barrier for the sideways movement of the polymer. The model does not impose self-avoidance on the polymer chain, but any move that would cross a gel strand is forbidden. The cage model is usually studied with a cage spacing of one, i.e., all edges of the cage lattice are occupied by gel strands. In this case the 90° moves are ruled out, since they always lead to the crossing of a gel strand. Evans and Edwards performed simulations with different cage spacings. In the remainder of this thesis, we limit ourselves also to a cage spacing of one.

Each move is on average attempted once per unit of time: a random monomer is chosen, and if this monomer is located in the middle of a kink, the kink will be given randomly one of six possible directions (its old direction or one of the five other ones). Thus, for the whole polymer, a total of $6L$ moves are attempted per unit time. The time per attempt is then given by $\Delta t = (6L)^{-1}$.

Computer simulations of the cage model resulted, at first, in anomalous scaling for the diffusion coefficient and relaxation time as a function of polymer length. Deutsch and Madden [10] reported $D \sim L^{-2.5\pm0.04}$ for the diffusion coefficient, and $\tau \sim L^{-3.41\pm0.14}$ for the tube renewal time. In 1998, however, Baumgärtner *et al.* [44] performed simulations with polymer lengths up to $L = 640$ monomers and $t = 10^7$ Monte Carlo time steps, and presented a thorough analysis of the time scales present in the polymer relaxation. They were able to confirm that the model showed the scaling of reptation theory, but large finite size effects existed. They used the Ansatz by Doi [45],

$$L^2 D = c\left(1 + kL^{-1/2}\right), \tag{2.1}$$

in which the finite-size effects of the diffusion coefficient D of a polymer with length L scales as $L^{-1/2}$.

In the same year, multispin coding techniques were introduced for the cage model [46]. This allowed to reach time scales close to two orders of magnitude longer. The diffusion constant was found to be $D \sim L^{-2}$, with strong indications that the leading exponent of the finite-size corrections has a value close to $-2/3$, in agreement with earlier findings in the repton model (see section 2.4).

2.4 Repton model

Another commonly used lattice model to simulate the dynamics of reptation is the so-called "repton model", introduced by Rubinstein in 1987 [47]. In this model, the polymer is described as a chain of L monomers, connected by

$N = L - 1$ bonds with either zero or unit length. Figure 2.4 gives an impression of a polymer in the repton model, and the gel surrounding the polymer. In the repton model, the zero-length bonds represent the defects described by De Gennes' model for reptation. The dynamics of the repton model consist of moves of monomers with one zero-length bond and one unit-length bond, plus moves of the end points.

An inner monomer in the repton model is in one of the following situations: (a) the monomer is in the same pore as both nearest neighbors; (b) both nearest neighbors are in adjacent pores; (c) the monomer has one nearest neighbor in the same pore and the other one is in an adjacent pore. In states (a) and (b) the monomer is unable to move. In state (c) the only allowed move is the move where the monomer joins its neighbor in the adjacent pore. The end monomers may be in one of two states: the nearest neighbor is either in an adjacent pore or in the same pore. In the first state, the monomer may join its neighbor in the adjacent pore; in the second state, the monomer is free to move to any of the six adjacent pores. In figure 2.5, monomers 0, 3, 5, and 10 may move to one other location, and monomer 11 may move to four new locations. All other monomers are unable to move. An elementary move consists of choosing a random monomer and trying to move it up or down the chain. For inner monomers each move is on average attempted once per unit of time. The time for one move is given by $\Delta t = (2L)^{-1}$.

One of the main advantages of the repton model, as compared to other polymer models, is the ease with which it can be projected onto a one-dimensional model. This projection, first introduced by Duke [48], is illustrated in figure 2.6. Deutsch and Madden [10] performed computer simulations of the one-dimensional repton model. For the diffusion coefficient, adjusted for the three-dimensional result, as a function of polymer length, they report a scaling of $D \sim L^{-2.3\pm0.1}$, and $\tau \sim L^{3.49\pm0.13}$. Barkema *et al.* [21] improved those results using multispin coding techniques to simulate the projected repton model. They obtained to leading order $L^2 D = 1/3$; analytic work on the model [49,50] yielded a proof for this numerically obtained result [51]. Barkema *et al.* also suggested that the dominant finite-size corrections are of the order $L^{-2/3}$; recent DMRG calculations [11,12] show clear indications that this suggestion is incorrect, and is only a good fit due to the combination of $L^{-1/2}$ and L^{-1} corrections, of which the first is dominant for long polymers.

2.5 Comparison of the models

Self-avoiding walks are an obvious choice as a model for polymers. One big disadvantage is, however, that self-avoiding walks with single-monomer moves

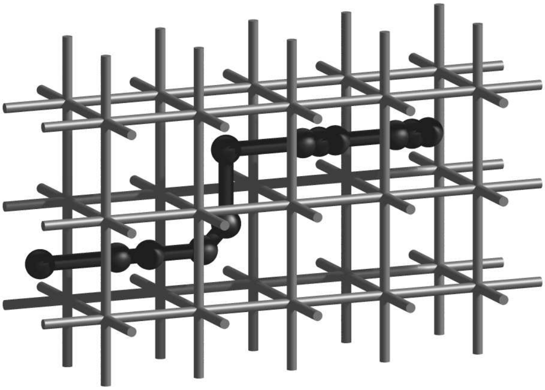

Figure 2.4: The three-dimensioal repton model. In the repton model, many monomers can occupy the same lattice site. For clarity of the figure, those monomers are drawn separately, but close to each other.

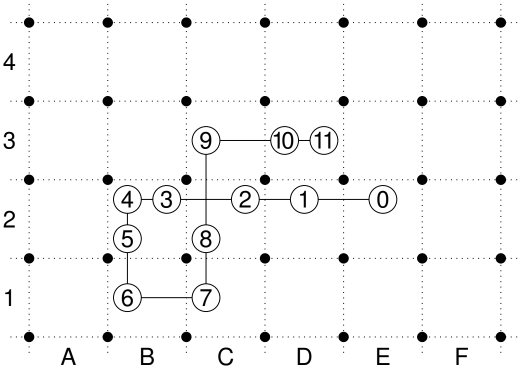

Figure 2.5: The repton model on a square lattice. The monomers in the polymer chain are connected by bonds of zero or unit length and each monomer is assigned to a face of the lattice, such that adjacent monomers in the chain occupy either the same or nearest-neighbor lattice sites. If a monomer is connected to one neighbor by a bond with zero length, and to its other neighbor by a bond with unit length, it can join its other neighbor. By doing so, the bond with zero length—a representation of "stored length"—diffuses along the chain. For instance, in the configuration depicted here, monomer 5 can join its neighbor 6. The ends of the polymer allow fluctuations of the total amount of stored length.

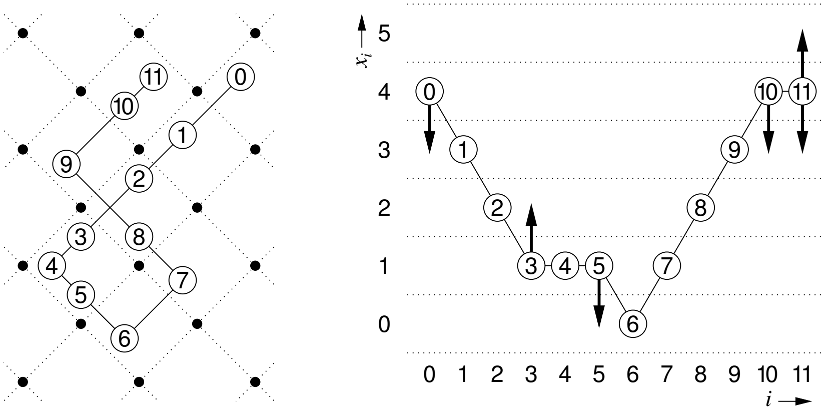

Figure 2.6: The two-dimensional repton model, rotated over 45°, can be mapped on a one-dimensional model, by storing for each monomer i only the displacement x_i, and disregarding the displacement in all directions perpendicular to that axes. By virtue of the simplicity of the dynamical rules in the repton model, the one-dimensional model can be simulated directly, without knowing the configuration before the projection. In the right figure, the allowed moves are depicted by arrows. The three-dimensional repton model can be mapped on the same one-dimensional model by choosing the x-direction of the model along the body diagonal of the cubic lattice.

have certain configurations from which no escape is possible, nor can they be reached from other configurations. Examples of such inaccessible configurations are sketched in figure 2.7 for two-dimensional self-avoiding walks. The configuration of the polymer shown in figure 2.7 (b) can be part of a longer polymer. Also in three dimensions such trapped configurations exist, but they are rare.

The bond-fluctuation model does not suffer from this severe lack of ergodicity, due to the precise choice of the allowed bond lengths [36, 37]. Its main disadvantage, however, is that the model does not lend itself for efficient computer simulation, due to the multiple sites occupied by each monomer, as well as due to the large number of elementary moves required for a significant configurational change.

The cage model suffers from a minor lack of ergodicity: if the lattice is divided into even and odd sites, analogously to a checkerboard, a monomer that resides initially on an even site will be located on even sites ever after. The model can be simulated highly efficiently, as we will show in chapter 4.

The repton model is ergodic, and can be simulated highly efficiently. Its main disadvantage is that the polymer dynamics is strictly limited to reptation, and Rouse dynamics is completely ruled out. However, in chapter 6, we will extend the model to include sideways movements.

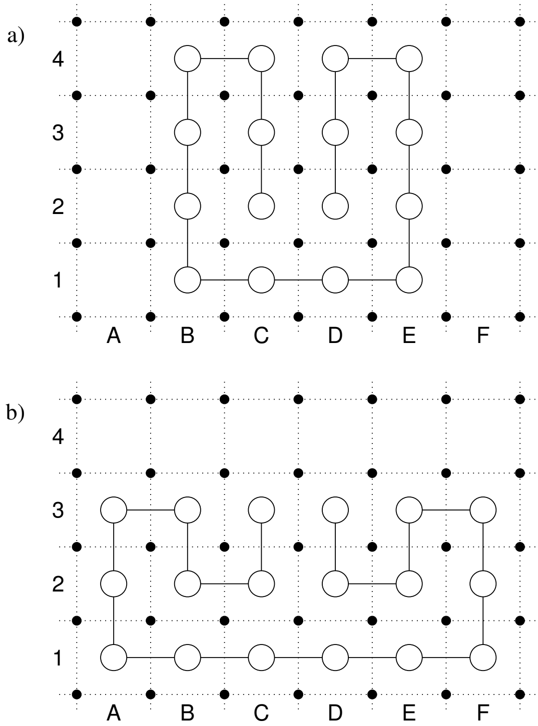

Figure 2.7: Self-avoiding-walk configurations which cannot be left if only single-monomer moves are allowed. The lower configuration can be part of a longer polymer. In combination with the pattern theorem, this means that infinitely long polymers will always have trapped segments. A polymer that is initialised in a trapped configuration, or contains a trapped subsegment, cannot diffuse macroscopically.

Although at first sight the cage model and the repton model show many similarities, one essential difference exists: the cage model allows for the formation of so-called *hernias* by accumulation of several kinks (see section 3.4, figure 3.1). Such hernias actually enable the interior of the polymer to select a new direction, later to be followed by one of the two tails. In experiments on gel electrophoresis, such hernias are found to occur [52, 53]. The cage model is in this respect more realistic than the repton model. With a slight modification to the repton model, it can also be used to model hernias: in the case where three neighboring monomers occupy one pore, the middle one should be allowed to move to a new pore and vice versa. This modification is usually left out because it prevents a projection of the repton model to a one-dimensional model, and the repton model with this modification is harder to treat theoretically.

Chapter 3

Gel electrophoresis

Gel electrophoresis is a widely used tool to separate mixtures of DNA molecules by length, as mentioned in section 1.5. In this chapter we explain how electrophoresis can be studied with lattice polymer models.

If the motion of a polymer can be approximated with overdamped dynamics of moves of small segments, and a small force is acting on every segment of the polymer chain, then the motion of the polymer is well described by a diffusive motion of the monomers which is biased in the direction of the applied force. We incorporate this into the lattice polymer models described in the previous chapter by assigning a charge q per monomer. We assume that the polymer is located in a homogeneous electric field $\vec{\mathcal{E}}$ that acts on the charged monomers. If the monomer is displaced over a distance $\vec{r}_{12} = \vec{r}_2 - \vec{r}_1$, the change in potential energy is given by $\Delta E = q\vec{\mathcal{E}} \cdot \vec{r}_{12}$, and the ratio of the corresponding Boltzmann probabilities is

$$P_1/P_2 = e^{\Delta E/k_B T} = e^{q\vec{\mathcal{E}}\cdot\vec{r}_{12}/k_B T}, \tag{3.1}$$

where k_B is the Boltzmann constant and T the temperature.

To incorporate the force acting on the monomers, we bias the forward an backward rates of the moves by the applied force. If equation (3.1) holds between each allowed Monte Carlo move and its reverse move, we say that the model adheres to *detailed balance*. The extended model then satisfies the following criteria: for a vanishing electric field it becomes equivalent to the basic model and the transition rates between the polymer states obey the criterion of detailed balance.

For electrophoresis, the most frequent choice of the actual rates allows only single-monomer moves and follows the so-called heat-bath rules: a monomer is removed from the chain and reinserted on a new location with a probability that is proportional to the local Boltzmann weight of the resulting configuration. If n orientations are possible, including the old one, then the probability to insert the monomer with orientation i is

$$P_i = e^{q\vec{\mathcal{E}}\cdot\vec{r}_i/k_BT} \Big/ \sum_{j=1}^{n} e^{q\vec{\mathcal{E}}\cdot\vec{r}_j/k_BT} = e^{(E_i-\bar{E})/k_BT} \Big/ \sum_{j=1}^{n} e^{(E_j-\bar{E})/k_BT}, \tag{3.2}$$

where $E_i = q\vec{\mathcal{E}} \cdot \vec{r}_i$ is the potential energy of the monomer in the applied field at position i, and $\bar{E} = \frac{1}{n}\sum_{i=1}^{n} E_i$ is the potential energy, averaged over the sites where the monomer can be reinserted. Note that the denominator of the last form of equation 3.2 is independent of the reference point of the potential energy. Therefore, we can measure time in units of $1/\sum_{i=1}^{n} e^{(E_i-\bar{E})/k_BT}$. The rate with which a move $j \to i$ is attempted is then simply $R_{ij} = e^{(E_i-\bar{E})/k_BT}$, independent of the previous position.

For models on a cubic lattice, we usually choose the direction of the electric field along one of the body diagonals of the unit cubes, because then the x, y and z directions are equivalent, and within one elementary move, the displacement r parallel to the field takes only two values: for the repton model $\pm 1/\sqrt{d}$ times the lattice spacing, and for the cage model $\pm 2/\sqrt{d}$ times the lattice spacing, where d is the dimensionality of the model.

3.1 Bond-fluctuation model

The two-dimensional bond-fluctuation model has recently been studied with an electric field acting on the polymers. Azuma and Takayama [54] added collective dynamics (sliding). Boileau and Slater [55] were able to compute numerically exact results for short polymers in a regular pattern of obstacles. The three-dimensional bond-fluctuation model has not been extended to simulate electrophoresis.

3.2 Cage model

In 1986, the cage model was extended by Olvera de la Cruz *et al.* [22] to include the driving force of an electric field along one of the main axes of the underlying lattice. This was done by accepting or rejecting proposed moves according to the Metropolis algorithm [56]. They simulated polymers in strong electric fields, and found that the relaxation time increases exponentially with increasing field strength. Their choice of the electric field strengths were, too strong to observe the fluctuation-dissipation regime, or even the plateau-mobility regime. Instead, the polymers are forced into U-shaped configurations, suspended from a gel fiber. In such configurations, the polymers may slide around gel fibers with some length-dependent friction coefficient. In 1991, Deutsch and Reger [23] performed computer simulations of the two-dimensional version of the model of Olvera de la Cruz *et al.*, in which they included long-range moves, describing the sliding of the polymer. They show that addition of long-range moves to the normal local moves is sufficient to change the debated exponentially long relaxation times of U-configurations. Instead they find a behavior equivalent to that of a continuum model presented earlier by Deutsch and Madden [57].

In order to mimic DNA electrophoresis, the electric field is applied in the direction of the body diagonal of the lattice. Moves which reinsert kinks pointing along with the electric field are attempted at an increased rate (P^+), and moves which reinsert kinks pointing against the field are attempted with a reduced rate (P^-), such that detailed balance is satisfied:

$$P^+ = \frac{1}{d} \cdot \frac{e^E}{e^E + e^{-E}} \qquad \text{and} \qquad P^- = \frac{1}{d} \cdot \frac{e^{-E}}{e^E + e^{-E}}, \tag{3.3}$$

where $E = q\mathcal{E}\sqrt{d}/(k_B T a)$, with a the lattice distance. The end monomers are always free to move. A new orientation for the end monomer is selected with probability P^+ in one of the d directions along the field or with probability P^- in one of the d directions opposite to the field.

We used multispin coding techniques to simulate the cage model in three dimensions. These simulations sample the regime where the fluctuation-dissipation theorem holds, as well as the regime of plateau mobility. In electric fields that are strong, but still much weaker than those used in earlier simulations, U-shaped configurations are observed with low drift velocities, decreasing strongly with the applied field strength.

3.3 Repton model

In 1989, the repton model was extended by Duke [48] in oreder to study DNA electrophoresis. This extended repton model became known as the Duke-Rubinstein model. The elementary moves again move a particle one unit of distance with or against the field. Just like in the zero-field model, in the simulation algorithm a random monomer is chosen. If this monomer has one nearest neighbor in the same pore and one in an adjacent pore, then the monomer is put with probability P^+ in the pore where the monomer has the lower potential energy and with P^- in the pore with the higher potential energy, such that we have a local thermal equilibrium, $P^+/P^- = e^E$:

$$P^+ = \frac{e^{E/2}}{e^{E/2} + e^{-E/2}} \qquad \text{and} \qquad P^- = \frac{e^{-E/2}}{e^{E/2} + e^{-E/2}}, \tag{3.4}$$

where $E = q\mathcal{E}\sqrt{d}/(k_B T a)$, with a the lattice distance. The end monomers are always free to move. An end monomer with a nearest neighbor in an adjacent pore may move to that pore with probability P^+ or P^-, and an end monomer in the same pore as its nearest neighbor gets a random new place with probabilities $\frac{1}{d}P^+$ and $\frac{1}{d}P^-$.

Monte Carlo simulations of this model show the fluctuation-dissipation behavior of equation (1.3) for small electric field strengths, as well as the presence of a plateau mobility for stronger electric fields [58, 59]. With multispin coding techniques, Barkema *et al.* [21] gave numerical evidence that the drift velocity in the regime of plateau mobility scales as $v \sim E|E|$. Comparing with the mobility in the fluctuation-dissipation regime, one finds that LE is a good scaling variable, since vL^2 is a function of LE in both regimes. It can be shown [60] that the drift velocity in the repton model decreases exponentially with E in the limit of high electric field strengths. Exact results for $E \to \infty$ show $v \sim e^{(2-L)E/2}$ and $v \sim e^{(3-L)E/2}$ for even and odd L, respectively.

3.4 Differences between the cage and repton models

Both the repton model and the cage model describe the motion of an entangled chain quite well. Monte Carlo simulations show similar behavior up to $LE \sim 10$, at least for chain lengths of up to several hundred monomers. For $LE \ll 1$ the drift velocity is proportional to the electric field. For $LE \sim 1$ a transition occurs between linear and quadratic behavior. In contrary to the repton model, the cage model allows for the creation of so-called hernias. A *hernia* is a build-up of stored length that protrudes from the confining tube of the polymer. We can show that hernias become important when the polymer is subjected to an electric field E_h, satisfying $L^{-1} \ll E_h \ll 1$. The difference between drift velocities computed with the repton model and the cage model in these relatively strong electric fields is due to polymer configurations with hernias.

The hernias play an important role in the motion of polymers [57, 59, 61–63]. Figure 3.1 shows how hernias cause the formation of U-shaped configurations. The net motion of the polymer requires the transport of kinks from the trailing end towards the head of the polymer. Thus, all kinks have to pass the hernia. The hernias tend to be oriented along the electric field so that kinks on the hernia have to travel against the electric field to get back to the backbone. Hernias, formed along the polymer chain, grow by capturing kinks. A consequence is that the distribution of kinks along the chain becomes uneven. Hernias occur in the cage model, but not in the repton model. As in the zero field case, the repton model may be changed so as to include the possibility of forming hernias, but then it loses its advantage in efficiency, since projection onto the one-dimensional model is no longer possible.

At the time of writing of this thesis, a converging picture emerges from all of

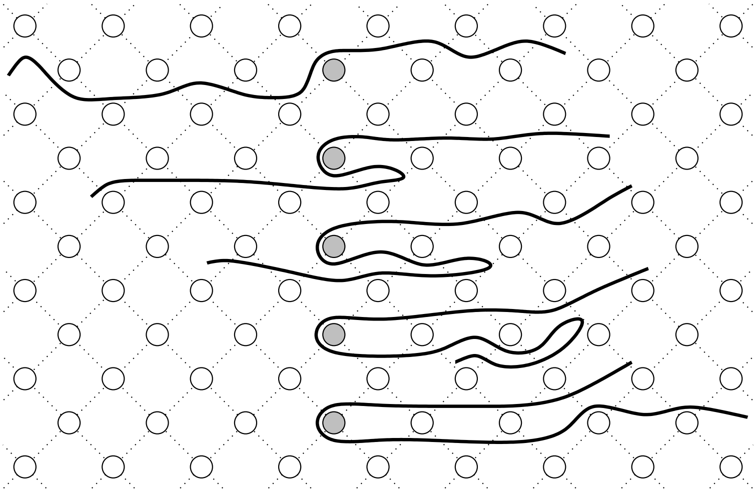

Figure 3.1: Dynamics facilitated by the existence of hernias. A moving polymer in a strong electric field is typically stretched parallel to the applied field (top picture). Sometimes a protrusion (hernia) develops, which is wrapped around one of the gel strands (the gray dot). In time, this hernia tends to grow, since it acts as a trap for stored length diffusing along the chain. Eventually, large parts of the chain, all the way to one of the end-points, can be "sucked" into the hernia. The final result is a U-shaped configuration with low mobility (bottom picture). Note that by construction, the repton model excludes this kind of dynamics, in contrast to the bond-fluctuation and cage models.

the lattice models discussed here. For a polymer confined in a gel, the dynamics is properly captured within the framework of reptation theory. The most accurate computer simulation results are obtained within the repton model. Here, the diffusion coefficient D as a function of polymer length L scales as $D \sim L^{-2}(1 + \mathcal{O}(L^{-1/2}))$. Qualitatively highly similar results are obtained for the cage model. Also, the bond-fluctuation model shows the same scaling of D, at least to leading order in L. The exponent of the leading finite-size corrections is still not determined accurately within this model. Combination of the scaling for the diffusion constant D as $D \sim L^{-2}$ with the fluctuation-dissipation theorem immediately provides the scaling of the drift velocity v with length and field strength as $v \sim E/L$.

The behavior changes once the electric field becomes sufficiently strong to influence significantly the typical shape of the polymer configuration. Once this happens, the regime of plateau mobility is entered, in which the drift velocity becomes independent of polymer length. This regime is experimentally well-known, and is also observed in each of the models discussed here. Cage model, repton model and biassed reptation model all yield scaling of the drift velocity in the regime of plateau mobility scales as $v \sim E|E|$, but in the biassed reptation model, fluctuations in the stored length along the chain should be properly accounted for.

For even stronger fields, the cage model as well as the repton model show an exponential decay of the drift velocity as a function of field strength. Likely however, this observed behavior is only a property of these models, and not related to experiment. It is argued in the literature that a description of the collective motion of a section of a polymer without stored length should be included in the lattice models by adding collective hopping of stored length to the allowed dynamics, once the field suppresses the density of stored length significantly [23, 54, 63, 64]. In our opinion, this issue is not yet resolved; a satisfactory understanding of the interplay between the DNA chain and the gel strand at the microscopic level is still lacking.

The agreement between the cage model and the repton model for all field strengths is highly remarkable, since there is a qualitative difference between these models. In the cage model, kinks can accumulate with the consequence that so-called “hernias” develop. These hernias tend to align with the electric field, and will trap stored length as it diffuses along the chain. Eventually, the whole chain will then fold into the hernia. This process is illustrated in figure 3.1. The repton model does not feature protrusions from the tube and never forms such hernias.

Chapter 4

Electrophoresis simulated with the cage model

In section 2.3 we described the cage model and in section 3.2 we described how this model can be extended to study electrophoresis. Here, we discuss a number of results obtained through an efficient implementation of the cage model, extended to describe electrophoresis. The details of this implementation are described in section 4.1. This efficiency turns out to be important, since the computational resources required increase tremendously with polymer length. In section 4.2 we report our results on the drift velocity of the polymers, and in section 4.3 we look at polymer shapes in different field strengths.

4.1 Multispin coding

As described in section 2.3, the cage polymer is viewed as a chain of monomers which are connected by bonds. Each bond can be oriented in $2d$ different ways, where d is the dimensionality of the simulated system. A polymer chain can be described by specifying the location of the first monomer and the orientation of all bonds. The advantage of this notation is that only the position of one monomer has to be stored plus the orientations of all bonds. The polymer in figure 4.1, for example, is described by the position of the first monomer, on the left side of the figure, and `+x-y+z+x-x+z`.

The dynamics can be described in terms of bonds. The bonds at the ends of the polymer are always free to change. The internal bonds are free to change only when they are part of a pair of oppositely oriented neighboring bonds (a kink). The first and last bond in figure 4.1 can change to any new bond: `+x`, `+y`, `+z`, `-x`, `-y` or `-z`. The kink configuration `+x-x` can change into any new kink: `+x-x`, `+y-y`, `+z-z`, `-x+x`, `-y+y` or `-z+z`.

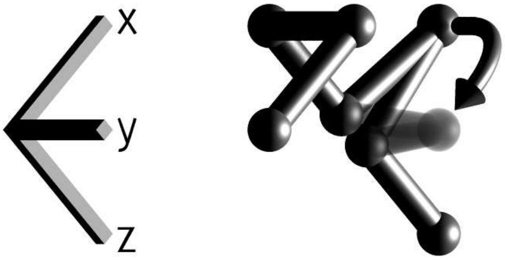

Figure 4.1: One elementary move of a monomer: a 'kink' (pair of anti-parallel neighboring bonds) is replaced by another kink.

	bond direction					
integer	`+x`	`+y`	`+z`	`−x`	`−y`	`−z`
$x^{(i)}$	1	0	0	0	1	1
$y^{(i)}$	0	1	0	1	0	1
$z^{(i)}$	0	0	1	1	1	0

Table 4.1: Encoding of a bond by three bits, where $x^{(i)}$ is the i^{th} bit of x and so on. Note that the encoding of a negative bond is the binary complement of the corresponding positive bond.

With multispin coding, many polymers can be simulated in parallel [46]. The idea is to write the most time-consuming parts of the simulation using only the logical instructions *and* ($\wedge$), *or* ($\vee$), *exclusive or* ($\oplus$) and *not* ($\neg$); since those instructions can be made to work on all the individual bits of an integer, each logical operation can be done for many polymers at once. Our implementation used 64-bit unsigned integers to simulate 64 different polymers in parallel. We encoded 64 bonds in three integers x, y and z. The three bits $(x^{(k)}, y^{(k)}, z^{(k)})$, contain the direction of the bond in polymer k, as shown in table 4.1.

An elementary move in the cage model for electrophoresis consists of a trial move of a randomly selected monomer i, with $0 \leq i < L$. If an inner monomer is selected, the two surrounding bonds are compared; if they are opposite, they are replaced by a randomly generated pair of opposite bonds. The first and last monomers are handled separately.

We can find which of the 64 polymers have a kink at monomer i by checking whether the surrounding bonds, $(x_{i-1}, y_{i-1}, z_{i-1})$ and (x_i, y_i, z_i), have opposite directions:

$$k_i = (x_{i-1} \oplus x_i) \wedge (y_{i-1} \oplus y_i) \wedge (z_{i-1} \oplus z_i). \tag{4.1}$$

Bit j of k_i is 1 if the surrounding bonds of the i^{th} monomer of polymer j are in opposite directions.

If a monomer can be moved, it will be relocated using a list of random kinks encoded in $\hat{x}$, $\hat{y}$ and $\hat{z}$. Bonds $i-1$ and i that surround monomer i are replaced by $\hat{x}$, $\hat{y}$ and $\hat{z}$ and their binary complements, respectively. Equation (4.2) shows how this can be done:

$$\begin{aligned} x_{i-1} &= (\neg k_i \wedge x_{i-1}) \vee (k_i \wedge \hat{x}) \\ y_{i-1} &= (\neg k_i \wedge y_{i-1}) \vee (k_i \wedge \hat{y}) \\ z_{i-1} &= (\neg k_i \wedge z_{i-1}) \vee (k_i \wedge \hat{z}) \\ x_i &= (\neg k_i \wedge x_i) \vee (k_i \wedge \neg\hat{x}) \\ y_i &= (\neg k_i \wedge y_i) \vee (k_i \wedge \neg\hat{y}) \\ z_i &= (\neg k_i \wedge z_i) \vee (k_i \wedge \neg\hat{z}). \end{aligned} \tag{4.2}$$

With only 27 logical operations the kinks at monomer i in all 64 polymers are replaced by new kinks, while polymers that have no kink at monomer i are left unaltered.

The first and last monomers are always free to move. If one of those is selected, we can just replace the bonds with randomly generated bonds: $x_0 = \neg\hat{x}$, $y_0 = \neg\hat{y}$, $z_0 = \neg\hat{z}$ if monomer 0 was selected and $x_{L-2} = \hat{x}$, $y_{L-2} = \hat{y}$, and $z_{L-2} = \hat{z}$ if monomer $L-1$ was selected. Equation (4.2), with $k_i = 1$, tells us that we have to use the binary complement of the random kink when monomer 0 is moved.

The complicating factor is that we need to keep track of the position of the first monomer. We have to calculate the distances travelled in the x, y and z directions. Since those directions are equivalent, we only calculate $r = x + y + z$. For this we only need to know whether the first bonds point at a positive direction, which is one of `+x`, `+y` and `+z`. This is done using the following equation:

$$d = x_0 \oplus y_0 \oplus z_0 \tag{4.3}$$

We do this both before and after we insert the random bonds. With this information we can calculate the new positions of the first monomers:

$$r_i = r_i + 2d^{(i)}_{\text{before}} - 2d^{(i)}_{\text{after}}. \tag{4.4}$$

This part of the simulation could not be efficiently implemented with multispin coding; we left it as a loop over all 64 polymers.

The algorithm described above relies on the availability of random kinks. These kinks should be generated with the probabilities as given in equation (3.3). Since the two bonds in a kink have opposite directions, only one bond has to be generated; the bond on the other side of the monomer is easily derived.

The properties of detailed balance are used to create those bonds correctly. Certain properties must be enforced: first of all the `x`, `y` and `z` directions should occur with the same probability; secondly the ratio of the probabilities for `+` and `-` bonds is given by quotient of P^- and P^+, as given in equation (3.3); this quotient is given by

$$P^{\mathrm{rel}} = P^-/P^+ = e^{-2E}. \tag{4.5}$$

The first property is enforced by rotating some of the bonds (we used 50%) the following way: $\mathtt{x}\mapsto\mathtt{y}$, $\mathtt{y}\mapsto\mathtt{z}$ and $\mathtt{z}\mapsto\mathtt{x}$. Using a randomly generated bit pattern r the following statements are used to rotate the bonds:

$$\begin{aligned} \tilde{x} &= (r \wedge \hat{x}) \vee (\neg r \wedge \hat{y}) \\ \tilde{y} &= (r \wedge \hat{y}) \vee (\neg r \wedge \hat{z}) \\ \tilde{z} &= (r \wedge \hat{z}) \vee (\neg r \wedge \hat{x}). \end{aligned} \tag{4.6}$$

The second property is then enforced by inverting some of the bonds. With 50% probability, the negative bonds are inverted and with P^{rel} times 50% the positive bonds are inverted. To make sure that all random kinks are independent we create a list of those and reshuffle this list regularly.

The simulation algorithm described above was implemented using the C programming language. We used a lagged (24, 55) additive Fibonacci random number generator. The simulations were done on a Silicon Graphics Origin 200 (180 Mhz) and on a DEC Alpha (466 Mhz) computer. The latter is faster and takes about 1.1 μs for 64 simultaneous Monte Carlo steps for $L = 100$. We have performed simulations for lengths up to 200. The polymers were initialized in a U-shape with both ends in the direction of the electric field. At regular intervals we checked whether the center of mass of a polymer had moved at least a distance equal to the the maximum distance between any two monomers in the direction of the applied field. When this occurred for a polymer, the measurement started. The measurement was stopped when all polymers had thermalized and the average distance travelled by all polymers was a few times their own size. We assumed that measurements are statistically independent when a polymer has travelled a distance equal to its own size. The CPU time taken to calculate the drift velocity varied from a few seconds for small polymers up to about 17 hours for the longest polymers ($L = 200$) in the smallest electric field ($E = 0.001$).

4.2 Drift velocity

The results of our simulations are presented in figure 4.2. The short polymers, up to length 20, show no superlinear dependence of the velocity on the electric field. When a small force, $E \ll L^{-1}$, is applied to the polymers, the velocity of the polymers increases linearly with the electric field. When a force around $E \approx L^{-1}$ is applied to the longer polymers, the polymer velocity increases

L	30	50	70	100	140	200
α	0.852	2.094	3.293	4.572	5.556	6.364
β	0.347	0.701	0.977	1.237	1.415	1.546

Table 4.2: Values for α and β, obtained by scaling the drift velocity $\tilde{v} = \alpha v L^2$ and electric field $\tilde{E} = \beta L E$, such that $\tilde{v} = \sqrt{\tilde{E}^2 + \tilde{E}^4}$. The values for α and β are used for scaling in figure 4.3. The values of α and β show convergence to a constant for large L.

faster than linearly with electric field. This dependence becomes quadratic for long polymer chains, as derived in section 1.5. For much larger electric fields, the velocity decreases to zero. For $E \sim 1$, it is known that the Monte Carlo approach is not realistic [57]. From figure 4.2, we see that the decrease sets in for much lower electric field strengths ($E_h \ll 1$). This is evidence that this velocity decrease is a real effect, not an artifact of this model.

As in the repton model [19], the drift velocity of long polymers shows a cross-over from linear to quadratic field-dependence. All graphs collapse on the function

$$\tilde{v} = \sqrt{\tilde{E}^2 + \tilde{E}^4}, \tag{4.7}$$

by scaling the drift velocities, $\tilde{v} = \alpha v L^2$ and the applied field strength $\tilde{E} = \beta L E$, see figure 4.3. The parameters α and β are expected to be constants for long polymers, but due to the strong finite-size effects we have fitted them for each value of L to get a good data collapse. The fitted values, shown in table 4.2, show convergence to a constant for large L.

4.3 Polymer shapes

For polymer length $L = 100$ we performed some short simulations to get insight in the typical movement of the center of mass of the polymer. In figure 4.4 the position of the center of mass, scaled with a factor of E^{-1}, is plotted as a function of time, for different field strengths. The starting positions of the polymers are chosen such that the graphs do not overlap. For the smallest electric fields the movement is just like one would expect from a diffusing particle, it moves randomly, but with some preferred direction. For electric fields in the middle range, the diffusion effect becomes relatively smaller. This results in a smoother behavior. In high electric fields the movement of the center of mass sometimes halts, when the force on the ends of the polymer pulls the polymer into a U-shape. When this happens, the polymer has to unhook itself before its center of mass can move forward again.

The polymer shapes in a small electric fields resemble those of a random

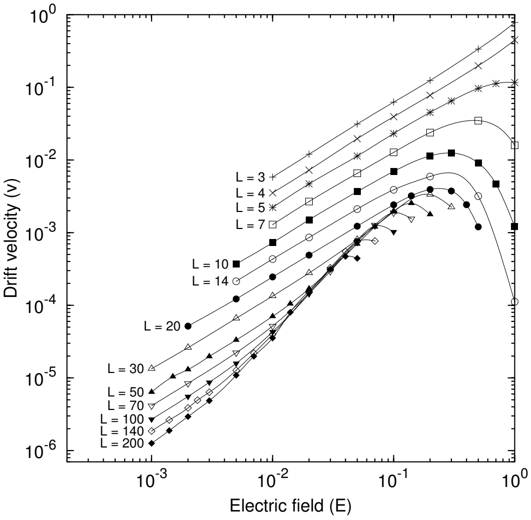

Figure 4.2: Drift velocity v of polymers of lengths L up to $L = 200$ in electric fields between $E = 0.001$ and $E = 1$.

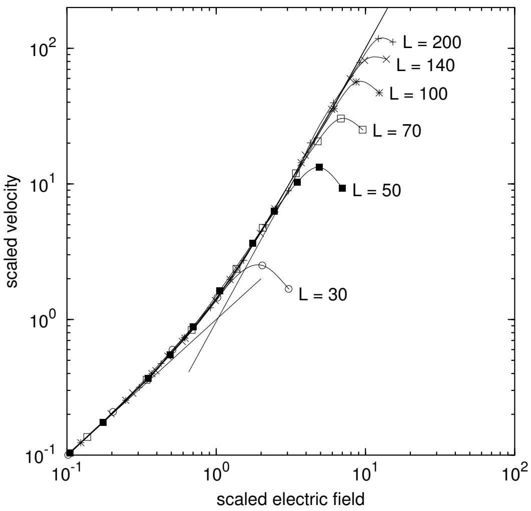

Figure 4.3: Transition between the linear and quadratic dependence of the velocity on the electric field. For various polymer lengths, the scaled velocity $\tilde{v} = \alpha v L^2$ is plotted as a function of scaled electric field $\tilde{E} = \beta L E$, such that they best fit the function $\tilde{v} = \sqrt{\tilde{E}^2 + \tilde{E}^4}$. For long polymers α and β converge to constants, see table 4.2. The straight lines indicate the linear and quadratic asymptotics of the scaled velocity.

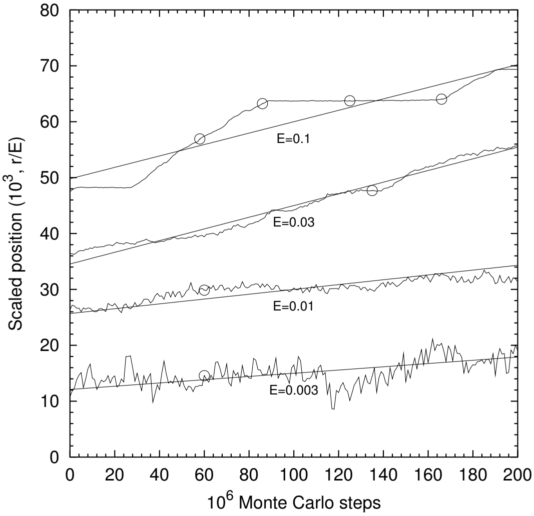

Figure 4.4: The position of the center of mass of a polymer as a function of the number of Monte Carlo steps. The sample polymer has length $L = 100$ and the position is divided by the applied electric field. The straight lines indicate the average velocity. From top to bottom, the electric field is $E = 0.1$, 0.03, 0.01, and 0.003. The circles denote the locations where snapshots are taken of the polymer configuration; these are shown in figures 4.5 and 4.6.

walk, as shown on the left side of figure 4.5. When the electric field is increased, the shape becomes stretched parallel to the electric field [65]; the configuration may be viewed as a set of blobs which move independently, as discussed in section 4.2. As shorter polymers move more quickly in a given electric field, the blob-configuration moves faster than a random walk configuration which results in a superlinear increase of speed as function of the electric field. When the electric field is increased above a certain value the shape may transform into a U-shape, as shown in figure 4.6. With higher electric fields it becomes more difficult to escape from this U-shape. Since the polymer cannot move sideways it is trapped in the lattice for a long time compared to the time it moves.

Figure 4.6 shows polymers in different configurations. The first polymer is stretched in the direction of the electric field. This configuration may be viewed as a large number of very small blobs. As such, the polymer has a high velocity, which may also be seen in figure 4.2 near $5.8 \cdot 10^7$ Monte Carlo steps. The second polymer is a transition configuration between the fast-moving cigar-like configuration as described above, and the U-shaped configuration. The polymer forms a hernia [57,63], which decreases the speed of the polymer locally.

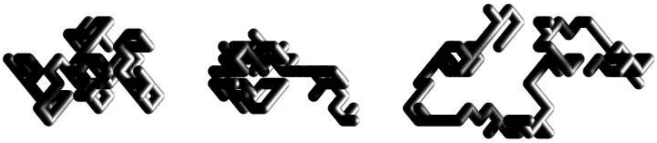

Figure 4.5: Three polymers of length $L = 100$ in different electric fields. From left to right: $E = 0.003$, 0.01, and 0.03. Polymers in small electric fields look like random walks; in slightly larger electric fields the ends tend to protrude. The center of mass displacement of these three polymers is shown in figure 4.4.

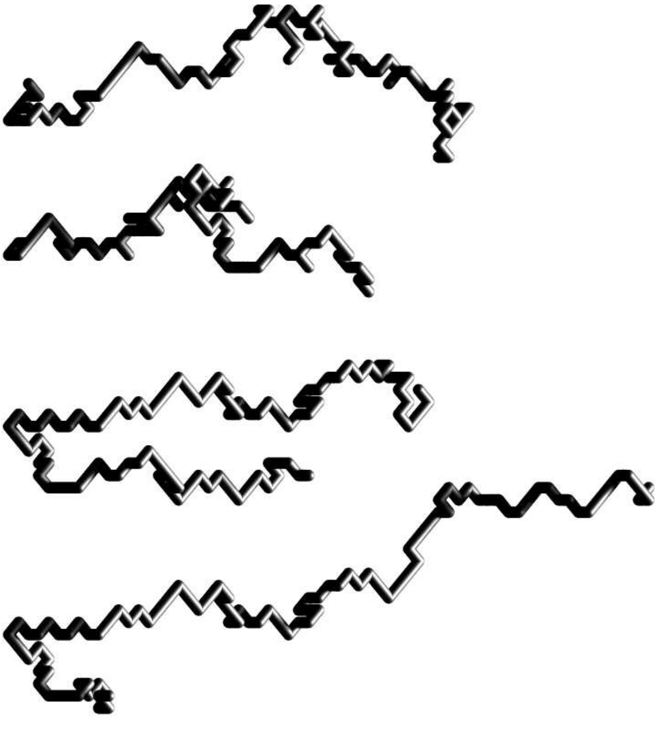

Figure 4.6: Snapshots of a polymer of length 100, in an applied electric field $E = 0.1$. In high electric fields the polymer does not look like a random walk, and the dynamics become complex. The center of mass displacement is shown in figure 4.4. From top to bottom the snapshots are taken at Monte Carlo steps: $5.8 \cdot 10^7$, $8.6 \cdot 10^7$, $1.25 \cdot 10^8$ and $1.66 \cdot 10^8$.

When the trailing end of the polymer passes the hernia, the third configuration appears. This polymer has a typical U-shape: the two ends both point into the direction of the electric field and much of the stored length diffuses out of the polymer. The motion of the center of mass stops, as can be seen in figure 4.2 near $1.25 \cdot 10^7$ Monte Carlo steps. The only way to escape from the U-shape is to create stored length at the shorter end of the polymer, and then transport it all the way against the electric field to the longer end. It takes a time exponential in the polymer length to escape from the U-shape [57]. Just before the polymer escapes from the U-shape, like in the fourth picture, its configuration is stretched and has almost no stored length. This state transforms quickly into a state that resembles the state of the first polymer in figure 4.6.

In weak electric fields, the polymer configuration is known to resemble a three-dimensional random walk. The average number of kinks is thus expected to be 1/6. For higher electric fields the U-shape configuration becomes more frequent. In this configuration the kinks are likely to diffuse towards the ends of the polymer, which means that the average number of kinks in the middle of the polymer decreases. When this happens we can no longer apply the blob argument as described in section 1.5. The mobility of the blobs in the middle of the polymer decreases as the average number of kinks in that region decreases. To check the dependence of stored length on the electric field we have performed some short simulations to measure the average number of kinks on each location along the polymer. The simulations consisted of 10^9 Monte Carlo steps after $2 \cdot 10^8$ steps of thermalization, starting with a random configuration. Every 10^6 Monte Carlo steps the kinks are counted. The fraction of time that a kink exists on a certain location is displayed in figure 4.7. Duke [58] showed that the chain of monomers in the repton model loses stored length, when subjected to electric fields. Here we find that the cage model shows a similar property. The average amount of stored length decreases in the middle of the polymer when the field strength is increased.

For higher electric fields, $E > L^{-1}$, the dynamics of the polymer becomes unstable: hernias are created along the polymer, which effectively reduce the number of kinks transported to the leading end of the polymer. This results in a lower velocity of the leading part of the polymer, while the velocity of the trailing end is not affected. The result is that the trailing end of the polymer will pass the leading end. This way, the polymers are likely to form the U-shaped configurations. In this configuration, both ends of the polymer point forward which results in a decrease of kinks near the base of the U-shape. Both effects are shown in figure 4.7 for polymers of length 100. For $E = 0.03$, the uneven distribution of kinks is clearly visible and for $E = 0.1$, the number of kinks in the middle of the polymer is clearly much lower than 1/6. When the density of kinks becomes less than 1/6 per monomer, the elastic force that contracts

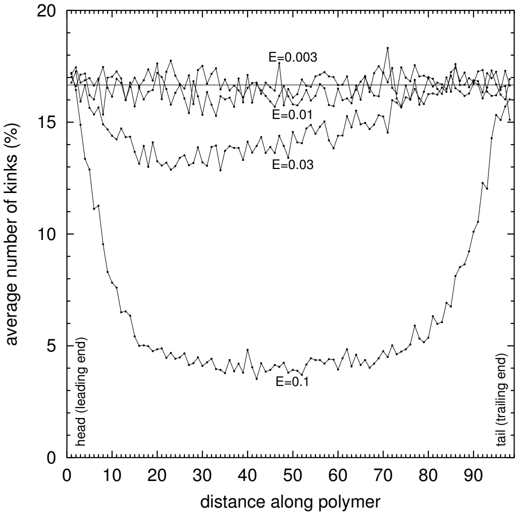

Figure 4.7: The average number of kinks as a function of the distance along the polymer; the end monomer that has the lower potential energy is the head of the polymer. The polymers are of length 100 and the electric fields are $E = 0.003$, 0.01, 0.03 and 0.1. The line gives the expected value 1/6 of kinks in a random walk.

the polymer is no longer in balance with the electric force. The polymer itself now transports the force along the chain, which may be better explained by the continuum model of Deutsch and Madden [57].

Chapter 5

Exact enumeration of the cage model for electrophoresis

In this chapter we present numerically exact computations on the cage model for electrophoresis, for polymers up to a length of $L = 15$. Numerically exact results can only be obtained for relatively small systems, because the number of configurations grows exponentially with the length of the polymer. For $L = 15$, we needed to exploit the symmetries in the model and to apply parallel processing techniques; without this, we could obtain only results up to a length of $L = 9$ in reasonable time. The simulations discussed in chapter 4 give results for much longer polymers, but they have inherently large statistical errors. Numerically exact results allow for a different class of analysis techniques, for instance those exploiting numerical differentiation.

The parallel implementation of the computation is done by spreading the nonzeros of the sparse transition matrix over the processors. Interprocessor communication is reduced by exploiting the specific sparsity structure of the transition matrix (see appendix A). The combined effects of decreasing the matrix size, improving the computational properties of the transition matrix, and applying parallel processing accelerates the computation by a factor of over a million. The results of the computations, combined with the results of the Monte Carlo simulations, are presented in section 5.5. The results relevant for computational science are presented in appendix A.

5.1 State space and the transition matrix

The cage model describes the dynamics of the polymers by giving the rates for directional change of kinks. A cage polymer of length L has $N = L - 1$ bonds where each bond points in one of six directions. Numbering the directions

`+x`$=0$, `+y`$=1$, `+z`$=2$, `-x`$=3$, `-y`$=4$, and `-z`$=5$ it is possible to enumerate the polymer states using $s=\sum_{n=0}^{N-1}6^{N-n-1}b_n$, where b_n is the direction of bond n in the chain. The transition matrix which represents the transition probabilities between polymer configurations has $6^{L-1}\times 6^{L-1}$ elements. This transition matrix T has $[5\,(\frac{L-2}{6}+2)+1]6^{L-1}$ nonzero elements: each polymer has $L-2$ inner monomers that can move if their bonds are in opposite directions, and two end monomers that can always move; a monomer that can move goes to one of five new positions or the polymer stays unaltered. The average number of elements per row is $\frac{5}{6}L+9\frac{1}{3}$.

If the monomer is moved along the applied field, bringing the polymer from configuration j to configuration i, then $T_{ij}=\delta t\cdot e^{E}$. If the monomer is moved against the applied field, $T_{ij}=\delta t\cdot e^{-E}$. The parameter δt can be interpreted as the time step in Monte Carlo simulations. The diagonal element T_{ii} is such that the sum of each column is exactly one. An upper bound for the sum of the off-diagonal elements in column j is $\delta t\cdot 3L\,(e^{E}+e^{-E})$, because at most L kinks can move, each in at most three forward and three backward directions. We choose $\delta t=(3L\,(e^{E}+e^{-E}))^{-1}$, so that all elements in column j are in the range $[0,1]$. Thus, T_{ij} is the transition rate to move from configuration j to i. Because we have a nonzero probability to stay in the same state, $T_{ii}>0$ holds for all i. This implies $T_{ij}<1$ for all $i\neq j$. Because end monomers can always move, we also have $T_{ii}<1$.

Applying the transition matrix to a state vector leaves the sum of the frequencies equal to one, by construction, since the probability of finding a polymer in any state is unity. The eigenvalues of the transition matrix are $1=\lambda_1>\lambda_2\geq\lambda_3\geq\cdots\geq\lambda_n>0$, for the choice of $\delta t=(3L\,(e^{E}+e^{-E}))^{-1}$.

The cage model with an applied electric field is ergodic, i.e., every configuration can reach every other configuration in a finite number of steps. The steady-state vector $\vec{a}$ is the eigenvector of T with eigenvalue one (which is the largest eigenvalue), normalized such that $\sum_i a_i=1$. The drift velocity of the polymer along one of the principal axes is

$$v=\frac{2}{3}\sum_i a_i(b_i e^{E}-f_i e^{-E}), \tag{5.1}$$

where b_i is the number of kinks and end monomers of polymer configuration i pointing backward (which can move forward with a rate of e^{E}), and f_i the number of kinks and end monomers pointing forward. The factor of $2/3$ appears because moves occur along each of the three principal axes, and because each kink move increases or decreases the sum of the coordinates of a configuration by two.

5.2 Power method

Repeatedly multiplying a starting vector by the transition matrix yields the steady-state vector. This iterative method to find the eigenvector of the largest eigenvalue is well-known as the power method, which converges if one eigenvalue is larger in absolute value than all the others. The speed of convergence of the power method depends on the ratio between the largest and the second largest eigenvalue.

Using repeated multiplication to find the eigenvector with eigenvalue $\lambda_1 = 1$ works if the absolute value of all other eigenvalues is smaller than one. The relative error in the solution decreases as $\mathrm{Max}(|\lambda_2/\lambda_1|, |\lambda_n/\lambda_1|)^k$, where k is the number of iterations performed. If we take $\delta t' = \omega(dL\,(e^E + e^{-E}))^{-1}$, the eigenvalues change to $\lambda_i^{(\omega)} = 1+\omega(\lambda_i - 1)$. This is equivalent to using the matrix $T^{(\omega)} = I + \omega(T - I)$, instead of T itself. The algorithm is still guaranteed to converge if $0 < \omega \leq 2$. The convergence is fastest if the smallest eigenvalue is the opposite of the next-to-highest eigenvalue: $\lambda_2^{(\omega)} = -\lambda_n^{(\omega)}$. We used $\omega = 2$, which works well in practice.

5.3 Reducing the number of states

In the model that we study here, the electric field is chosen in the $(1,1,1)$ direction, and consequently polymer configurations that are related through rotation of $2\pi/3$ around the direction $(1,1,1)$ are equivalent, i.e., their probability is the same, irrespective of the field strength. Moreover, in many cases it is possible to rotate a *part* of the polymer around this direction while preserving this equivalence. Another symmetry that results in different polymer states with identical frequency is due to the order of numbering the steps. To enumerate a polymer configuration, one has to start at one of the two end points of the polymer, which gives two possibilities to number the state, except if the state is symmetric.

If polymer configurations are grouped into classes containing only equivalent polymers, it is sufficient to determine the probability for one polymer configuration per class rather than for all polymer configurations, since by definition the probabilities are equal within a class. Since equivalence classes can easily contain thousands of configurations, the state space is thus reduced by several orders of magnitude, and a tremendous speedup is obtained.

To identify which polymer configurations are equivalent, we construct a representation that puts equivalent configurations in the same class. We call part of a configuration between two monomers *removable* if all monomers between them can be removed by repeatedly removing kinks. A kink is removed by delet-

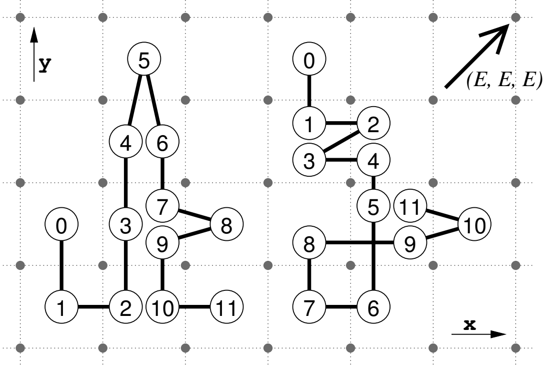

Figure 5.1: The cage model. The dotted lines denote the gel strands in the x-y plane, and the large gray dots are the gel strands in the z direction. The space between the gel strands represents the pores of the gel. The polymer is modeled as a chain of monomers; two adjacent monomers reside in nearest neighbor pores. We denote bonds that are going right, left, up, down, out of the paper, and into the paper by `+x`, `-x`, `+y`, `-y`, `+z`, and `-z`, respectively. The two example configurations were chosen to be planar, for clarity. The electric field vector points diagonally out of the paper.

ing the central monomer and the two bonds connected to it, and merging the two monomers adjacent to the central monomer. In the left part of figure 5.1, monomers 4 and 6 are merged when the kink at monomer 5 is removed. If two polymers have the same sequence of forward and backward bonds, and the same set of removable pairs of monomers, they have the same *kink representation.* The construction of such a representation is illustrated in figure 5.2. The kink representation assigns a unique number to each symmetry class. We can prove that all polymer configurations with the same kink representation have the same probability in the steady state (see appendix B) and, as a check on the algorithm, we have verified this explicitly up to $L \leq 9$. Furthermore, the forward/backward symmetry was removed by also computing the kink representation starting at the other end of the polymer, and then using only the one with the lower binary value.

The reduced state space is constructed by computing the kink representation for each polymer configuration and removing the duplicates. During this phase some additional information is stored about each kink representation: each bond representation that introduces a new kink representation is stored along with the kink representation, and the total number of bond representations for each kink representation is recorded. Table 5.1 shows the reduction of the configuration space obtained by removing the symmetries. The kink representations are enumerated by sorting them based on their binary value, with the rightmost bit

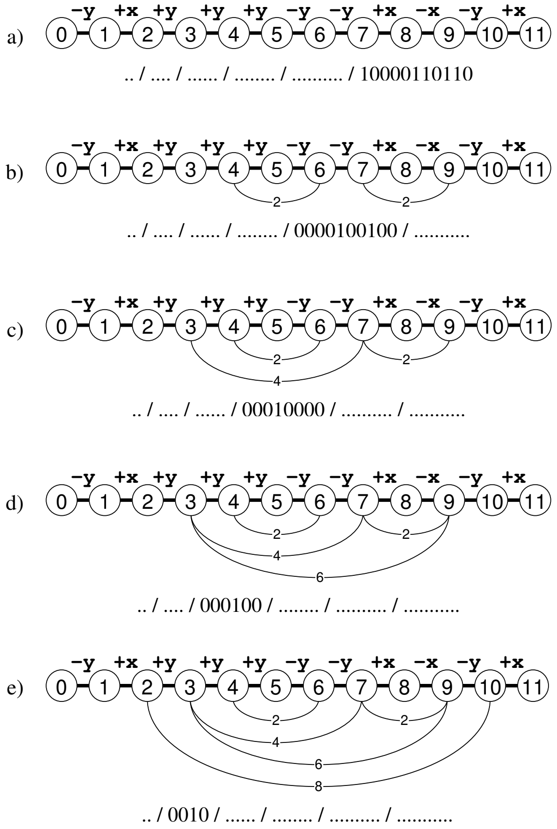

Figure 5.2: Kink representation for the polymer configuration in the left of figure 5.1. a) First, walking along the polymer, for each step we assign either a 0, in case the step is taken along the field, or 1, in case the step is taken against it. Thus, bonds `-x`, `-y`, and `-z` translate into 1, while bonds `+x`, `+y`, and `+z` translate into a 0. b-e) Iteratively, removable parts are identified. In the polymer configuration, these removable parts look like "Christmas trees". The smallest tree is a single kink. Larger trees show a wide variety of structure. If a successive removal of branches (kinks) leads to a complete removal of the segment, it corresponds to such a "Christmas tree", otherwise not. In more detail, the procedure is as follows: b) From left to right, subsegments of length 2 are searched for kinks. In the binary notation a kink is represented by a 1 and no kink by a 0. c) Subsegments of length 4 are checked whether they are removable by successively removing kinks. Note that only subsegments that start and end in the same lattice site can be removable, but subsegments that form loops (like `+x+y-x-y`) are not removable. d) and e) show the same for subsegments of lengths 6 and 8. The concatenation of the binary representations resulting from the bond orientation and the removability form a binary number that uniquely identifies the symmetry class of the polymer.

L	kink representations	reduction factor	nonzero elements	reduction factor
3	5	7	19	22
4	9	24	49	56
5	37	35	233	75
6	93	84	785	142
7	340	137	3 084	229
8	1 015	276	11 003	407
9	3 534	475	41 594	680
10	11 397	884	150 645	1 182
11	39 082	1 547	559 722	1 999
12	130 228	2 786	2 032 536	3 451
13	445 315	4 888	7 479 343	5 869
14	1 505 785	8 674	27 130 349	10 110
15	5 154 859	15 202	99 199 551	17 248

Table 5.1: The number of kink representations for polymer lengths $L = 3$–15, the reduction factor of the state space, the number of nonzero elements for the matrix in the kink representation, and the reduction factor of the number of nonzero elements.

the least significant. This ordering has the property that in most cases moves cause only small changes in binary values, e.g., replacing a kink `+x-x` with `-y+y` swaps two bond-direction bits; replacing `+x-x` with `+y-y` even keeps them the same. The removable-parts bits can be affected as well, but this becomes less likely with increasing part length.

5.4 Reduced transition matrix

The reduced transition matrix T' is constructed one column at a time. For column j, we consider the possible moves of the bond representation stored with kink representation j. For each resulting bond representation, we compute the associated kink representation i, concluding that kink representation j can move to kink representation i in a single-monomer move, and the reduced transition matrix element T'_{ij} is incremented by either $\delta t \cdot e^{E}$ or $\delta t \cdot e^{-E}$, depending on the direction of the move. Table 5.1 shows the resulting reduction factor in the number of nonzero matrix elements. By construction, the sum of the elements in a column of the matrix T' is one. All elements are in the range $[0, 1)$, and the diagonal elements T'_{ii} are in $(0, 1)$. The reduced model is ergodic, and its steady-state vector is the normalized eigenvector with eigenvalue one (which is the largest eigenvalue). The drift velocity is again computed by equation (5.1),

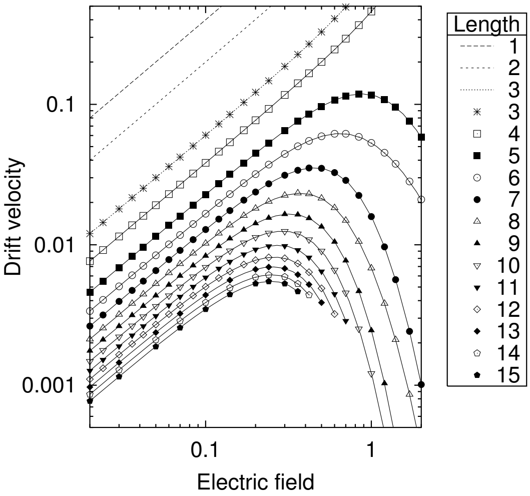

Figure 5.3: The graphs show the computed drift velocities of the cage polymers as a function of electric field strength E. For $E \leq 1$, the relative error is less than 10^{-10}; all other points have a relative error less than 10^{-4}. The graphs for lengths 1, 2, and 3 are $v_1 = 2(e^E - e^{-E})$, $v_2 = e^E - e^{-E}$, and $v_3 = 4(e^{3E} - e^{-3E})/(18 + 11(e^{2E} + e^{-2E}))$ respectively; for $L > 3$, the computed points are connected by straight lines.

where a_i is now the probability of class i, and b_i (f_i) the number of backward (forward)-pointing kinks and end monomers of a single polymer configuration in class i.

5.5 Results

Figure 5.3 shows the numerically exact values for the drift velocity of the cage polymers up to length $L = 15$. Drift velocities for $E \leq 10^{-2}$ have been omitted. For weak electric fields a linear scaling with the electric field strength is expected. Indeed, the graphs of all polymer lengths show a linear scaling with the electric field, as long as the electric field is well below L^{-1}. For higher electric fields, polymers of lengths $L = 2$, 3, and 4 show an increase in velocity, while longer polymers show a decrease in velocity. The difference in behavior of the short polymers, up to length 4, and the longer polymers is because the polymers need at least 5 monomers to feature a so-called U-configuration. In these U-configurations, the polymer is folded over a gel strand, while the electric field pulls on both ends, thereby stretching the polymer.

In the Monte Carlo data from the previous chapter, figure 4.2, the regime just before the maximum velocity where the drift velocity increases quadratically

L	D	L^2D
3	0.200 000 000 000	1.800 000 000 0
4	0.095 541 401 266	1.528 662 420 3
5	0.045 892 037 845	1.147 300 946 1
6	0.028 134 332 038	1.012 835 953 4
7	0.018 844 680 457	0.923 389 342 4
8	0.013 302 014 727	0.851 328 942 5
9	0.009 776 090 804	0.791 863 355 1
10	0.007 424 928 047	0.742 492 804 7
11	0.005 790 292 327	0.700 625 371 6
12	0.004 615 107 027	0.664 575 411 8
13	0.003 746 569 186	0.633 170 192 5
14	0.003 089 624 043	0.605 566 312 4
15	0.002 582 785 984	0.581 126 846 5

Table 5.2: Diffusion coefficients for cage polymers up to length $L = 15$, obtained by a second order polynomial fit to the mobility for field strengths in the range $[10^{-6}, 10^{-3}]$. An upper bound for the relative error is 10^{-9}. For long polymers, L^2D converges to a constant [46].

with the field strength is clearly visible. It is not visible for the relatively short chains for which we can compute numerically exact data.

The diffusion constant D is retrieved from the data of the drift velocities by the Nernst-Einstein relation $v = (D/k_BT)F$, valid in the limit of a small force F, where the drift velocity scales linearly with the electric field. From the graphs of the drift velocity, figure 5.3, it is clear that the linear regime is found. We used the computed velocities in the range $E = 10^{-6}$ to 10^{-3} to fit a second-order polynomial to the mobility $\mu = v/E$ of the polymers, to obtain the value of the diffusion constant at $E = 0$ (see Table 5.2). The standard deviation of the fit to the mobility was about 10^{-9}.

It is known that asymptotically for large polymers the diffusion coefficient behaves as $D \sim L^{-2}$, but with large finite-size corrections for usual polymer lengths. As the polymers are modeled as a random walk of $N = L - 1$ steps, finite-size corrections of the order of $N^{-1/2}$ are expected. Let us call $d(N) = D \cdot (N + 1)^2 = DL^2$, and $d_\infty = (DL^2)_{L\to\infty}$; we expect that for large but finite polymers $d(N) = d_\infty + aN^{-x}$. The parameters a and x can be found from this equation by differentiation: $\frac{\partial d}{\partial N} = -axN^{-x-1} \approx \frac{1}{2}(d(N + 1) - d(N - 1))$. A least-squares fit of the derivative of the new data against N for $N = 8$–13, shown in figure 5.4, gives $a = 2.469(5)$ and $x = 0.512(6)$, strongly suggesting finite-size corrections with an exponent $\frac{1}{2}$. This shows an advantage of the numerically

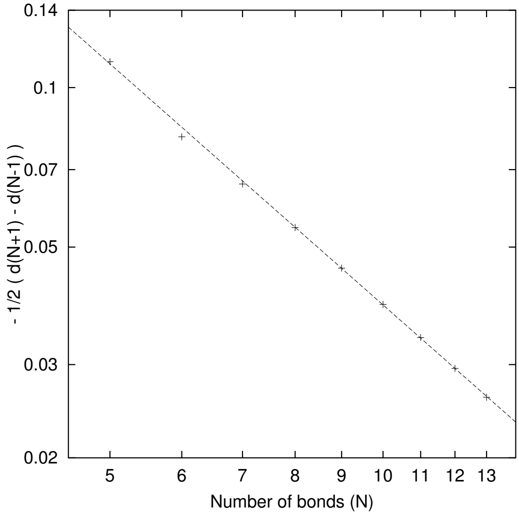

Figure 5.4: Finite size corrections of the cage model for electrophoresis. For short polymers, the diffusion coefficient deviates from $D \sim L^{-2}$. Numerical differentiation of DL^2 shows that the deviation from the asymptotic behavior behaves as the inverse square root of the polymer length.

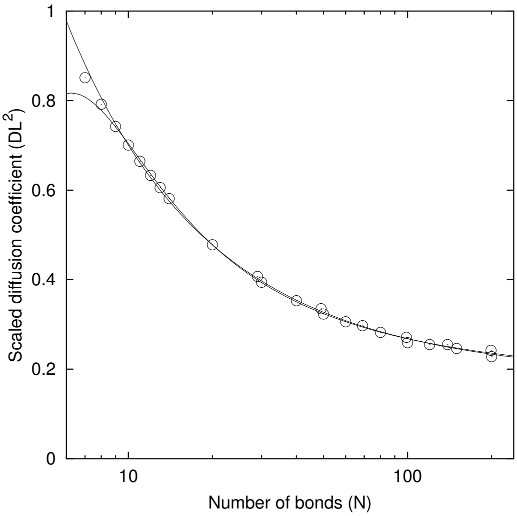

Figure 5.5: A fit of the functions $a + bN^{-1/2} + cN^{-1}$ and $(a' + b'N^{-1/2} + c'N^{-1})^{-1}$ to $D(L)L^2$, where $D(L)$ is the diffusion coefficient of the cage model of electrophoresis for a polymer with L monomers. The diffusion coefficients for polymers of length $L \leq 15$ are numerically exact, while the other points are computed from Monte Carlo simulations.

exact computations over Monte Carlo simulations, in that we can compute the derivative of the data reliably.

We used our new diffusion coefficients, combined with data from refence [46], and the results for small electric fields from section 4.2, to find the length dependence of the diffusion coefficient. Figure 5.5 shows the result of a least-squares fit with $d(N) = a + bN^{-1/2} + cN^{-1}$, which gives gives $d(N) = 0.172(6) + 0.63(8)N^{-1/2} + 3.3(2)N^{-1}$, and a least-squares fit with $d(N) = (a' + b'N^{-1/2} + c'N^{-1})^{-1}$, which gives gives $d(N) = (5.67(5) - 22.2(5)N^{-1/2} + 28(2)N^{-1})^{-1}$. Both of these expansions converge, within the error margins, to the same value for large N. The first expansion converges to 0.172(6), and the second expansion converges to $1/5.67(5) = 0.176(2)$. Combining these results, we conclude that for large L the diffusion coefficient is $D = 0.175(2)L^{-2}$. Our diffusion coefficient agrees to leading order with that of Barkema and Krenzlin [46], but they reported a different finite-size scaling: $DN^2 = 0.173 + 1.9N^{-2/3}$.

Chapter 6

Extensions of the repton model for simulating melts

This chapter discusses the extensions of the repton model which are required to simulate the dynamical properties of dense polymer solutions and melts. Since the dynamics of polymeric systems occur on long time scales and large length scales, computational efficiency is of paramount importance. We therefore discuss in detail how the resulting extended repton model can be implemented with high computational efficiency, exploiting multispin coding.

With our extended repton model, we simulated the phase separation of a dense binary polymer mixture. We performed simulations of a system with 46 080 polymers, each containing 100 monomers, located on a lattice with about 14 million sites. We verified that over the final two decades in time, the domain size $d(t)$ grows according to $d(t) \sim t^{1/3}$, as expected for a system with overdamped dynamics and a local conservation law. This simulation involved $3.4 \cdot 10^{13}$ elementary moves, but could nevertheless be carried out on a single-processor workstation in about 20 days, which demonstrates the usefulness of the approach described here.

6.1 Repton model for polymer melts

The repton model as proposed by Rubinstein, introduced in section 2.4, simulates a single polymer obeying random-walk statistics, with dynamics limited to the diffusion of stored length. Two extensions of the repton model are needed to use it for the simulation of entangled chains: we have to impose excluded volume constraints on the chains and we have to allow sideways motion of the interior monomers of the chains.

The repton model describes "stored length" as two or more monomers on the same lattice site. Therefore, we cannot restrict the polymers to pure self-avoiding-walk statistics disallowing two monomers to occupy the same lattice site, without destroying the reptation dynamics that so successfully described polymer-diffusion in a gel. As reptation is also expected to play an important role in the dynamics of a polymer in a melt, we should keep this kind of dynamics. A good way to combine self-avoidance with reptation dynamics is to limit only the trajectories of the polymers to self-avoiding-walk statistics. The set of lattice sites occupied by the polymer is called the *tube*, and the difference between L and the length of the tube is the amount of stored length. Monomers belonging to different polymers are never allowed to reside on the same lattice site; a lattice site is allowed to contain two or more monomers, but only if those monomers are adjacent along the chain. One convenient side-effect of this choice is that reptation moves do not affect the tube, so they never cause a violation of volume exclusion.

Sideways motion is implemented by also allowing monomers to move by a single lattice spacing to empty sites outside the tube if such a move leads to another allowed tube configuration. In particular, from a multiply occupied site, only the monomers with a neighbor in another site are allowed to move sideways, as moves of the other monomers lead to a forbidden tube configuration. Single-monomer moves work well on lattice structures that contain loops of three sites, like the triangular and the face-centered-cubic (FCC) lattice; on lattices without such three-site loops, like the square and cubic lattices, the sideways movement can only be implemented by allowing multiple bond lengths between adjacent monomers.

Figure 6.1 shows model-polymers on a triangular lattice. In the upper polymer in the figure, reptation moves are possible for monomers 2, 4, 6, 9, 10, and 11; sideways moves can be made by monomers 2, 4, 6, 8, 9, 10, and 11. Monomers 3, 5, and 7 cannot move. The end-monomers 1 and 12 can move to any empty nearest-neighbor site. In the lower polymer, reptation moves are possible for monomers 3, 5, 6, 10, and 11; sideways moves can be made by monomers 3, 4, 5, 6, 7, 10, and 11. Monomers 2, 8, and 9 cannot move. End monomer 1 can move to any empty nearest-neighbor site, while end monomer 12 can only move to the site occupied by monomer 11. Sideways moves of a monomer with both its neighboring monomers in the same lattice site, such as monomer 5 of the upper polymer and monomer 2 in the lower polymer, are not allowed, because after such a move the trajectory of the polymer is no longer self-avoiding. Sideways moves of end monomers are not allowed, so monomer 12 of the lower polymer can only move to the site containing monomer 11 of the same polymer.

This particular lattice polymer model lends itself extremely well for computer simulations. The polymers diffuse by two different kinds of Monte Carlo

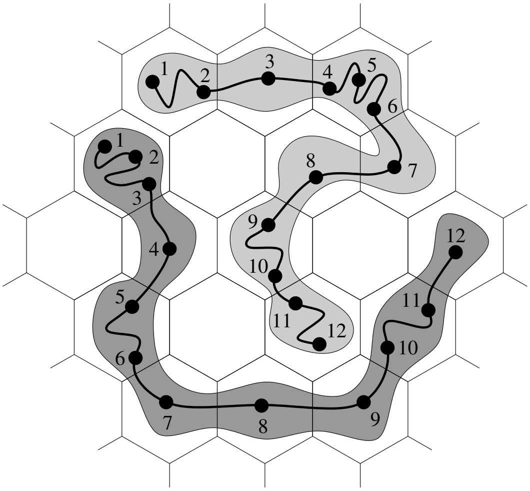

Figure 6.1: Extended repton model in two dimensions. The excluded volume constraint is implemented by allowing only adjacent monomers in a polymer to occupy the same lattice site. Phase separation is induced by nearest-neighbor lattice site interactions.

mechanisms. The first mechanism is that of reptation along a chain: if for a given monomer one of its adjacent monomers is located on the same lattice site, while its other adjacent monomer is located on a nearest-neighbor lattice site, it can move to the latter. As we will show below, interactions can be defined such that this kind of move does not change the total energy. This allows a highly efficient implementation of those moves using multispin-coding techniques [66].

The effective rate of reptation depends on the amount of stored length. If all moves were performed with equal probability, the amount of stored length would depend on the lattice coordination number z. To keep the amount of stored length comparable with that of the one-dimensional repton model, we have chosen a ratio of $z/2$ between attempt rates for moves that increase or decrease the tube length. Here z is the lattice coordination number, which is 12 for the FCC lattice. If no self-avoidance were imposed, this would lead to a stored-length density of 1/3, as in the one-dimensional repton model. From a physical point of view, one can see this as a computational tool to bring the amount of stored length in agreement with what would be observed experimentally.

6.2 Implementation of the projected repton model

Multispin coding is a programming technique which makes use of the low-level bit manipulation functions provided by the CPU of our computer, effectively doing many simple computations in parallel. Here, we will explain this technique analogously to reference [21], as applied to the projected repton model. Next we will discuss the implementation of this technique in our model for polymer melts. For a general introduction to multispin coding we refer to reference [66], Chapter 15. The low-level functions used to write multispin-coded simulations include logical operations like AND ($\wedge$), OR ($\vee$) exclusive-or (XOR, $\oplus$), negation (NOT, $\neg$), as well as bitwise shifts ($\ll$, integer multiplication by a power of 2; $\gg$, integer division by a power of 2); and arithmetic operations like addition and multiplication.

In the projected repton model, the state of the chain with monomers $i = 1 \dots N$ is specified by the set of their coordinates $\{x_1, \dots, x_N\}$. Monomers which are adjacent along the chain have to reside either on the same lattice site, or on nearest-neighbor sites, with the consequence that the step $s_i \equiv x_{i+1} - x_i$ can only take the values ± 1 or 0. An alternative way to store the polymer state is therefore to store the coordinate x_1 of the first monomer, and to store for step $i = 1 \dots N-1$ two arrays of bits (A_i, B_i) that can take the combination $(0,0)$ if $s_i = 0$, $(0,1)$ if $s_i = 1$, and $(1,0)$ if $s_i = -1$.

In one unit of time, each monomer on average attempts to move once in each direction. Thus, $2N$ elementary moves, i.e., one monomer attempting to move in one of the directions, should be attempted per unit of time.

If monomer i in the interior of the chain can join its neighbor $i+1$, it cannot join its neighbor $i-1$ (because it is on the same site already), and vice versa. We exploit this by attempting these two elementary moves simultaneously. The first and last monomer can move in either direction, if they are located in the same site as their neighbor along the chain, so this trick cannot be applied to these two monomers. If the dynamics of the model is studied, the end monomers should therefore be selected twice as frequently as an interior monomer.

In detail, each monomer in the interior of the chain is selected with a probability $1/(N+2)$. If the selected monomer i is located in the same site as exactly one of its two adjacent neighbors, it will join the other neighbor. Whether a move of an interior monomer i is possible is computed from the left and right bonds of monomer i: (A_j, B_j) and (A_i, B_i), with $j = i-1$. A move is possible if one of the two pairs is $(0,0)$ and the other is not. If the move is carried out, (A_i, B_i) and (A_j, B_j) are exchanged. A sequence of logical operations that achieves this is:

$$\begin{array}{rcl} y & = & (A_i \vee B_i) \oplus (A_j \vee B_j) \\ m_A & = & (A_i \oplus A_j) \wedge y \\ m_B & = & (B_i \oplus B_j) \wedge y \\ A'_i & = & A_i \oplus m_A \\ B'_i & = & B_i \oplus m_B \\ A'_j & = & A_j \oplus m_A \\ B'_j & = & B_j \oplus m_B \end{array} \tag{6.1}$$

The first and last monomers are selected, each, with a probability $2/(N+2)$; twice the probability to select a specific monomer in the interior of the chain. After the selection of an end monomer, the intended direction is also randomly selected, with 50% probability to be in the positive or negative direction. If the first monomer is selected, the following statements can be used to update bond (A_1, B_1), trying to displace the end monomer in the negative or positive direction based on the value of r_1.

$$\begin{aligned} A'_1 &= \neg(B_1 \vee r_1) \\ B'_1 &= (\neg A_1) \wedge r_1 \end{aligned} \tag{6.2}$$

If the last monomer is selected, similar statements suffice to update bond (A_{N-1}, B_{N-1}). However, in case the first monomer is selected, also its coordinate x_1 needs to be updated. Since this coordinate is an integer number that can take a wide range of values, its update is not implemented in a multispin coding fashion. Luckily, the probability to select the first monomer decreases with increasing polymer length.

The motivation for multispin coding lies in its efficiency. The simulation of the dynamics of 64 polymers involves 128 elementary moves per time unit, per monomer. With this multispin-coding implementation, only 11 logical operations, 4 loads, and 4 stores are needed to perform those 128 elementary moves. On a fast workstation, this takes 41 ns of CPU time, or 0.32 ns per elementary move. For updating the end monomers, even fewer logical operations suffice. For the first monomers, however, the update of the 64 values of x_1 cannot be achieved as efficiently, and a loop over all 64 polymers is inevitable, with the consequence that the simulation of the first monomer requires 5.8 ns CPU time per elementary move.

The multispin-coding implementation outlined above will thus perform 64 simulations in parallel. They are correlated, however, since they share the sequence of selected monomers. In fact, if at some point two simulations are in identical polymer configurations, they will stay in identical configurations ever after; in long simulations of small systems, this "locking" will inevitably happen. Complete locking is avoided by an uncorrelated choice in the directions in which end monomers attempt to move, using a 64-bit random bit pattern rather than a binary choice between all-up or all-down. If desired, more de-locking can be obtained at the expense of a lower efficiency, by rejecting a fraction of the allowed moves, also using a 64-bit random bit pattern. An important remark is, however, that as long as each of the 64 simulations is correct in itself, one obtains 64 unbiased results; one should just be careful in assigning significance to the spread in those 64 simulations.

vector in Z^4			Z^3
$(-1,\ 1,\ 0,\ 0)$	$\equiv$	$\hat{t}$	$(-1,\ 1,\ 0)$
$(\ 0,\ -1,\ 1,\ 0)$	$\equiv$	$\hat{u}$	$(\ 0,\ -1,\ 1)$
$(\ 0,\ 0,\ -1,\ 1)$	$\equiv$	$\hat{v}$	$(\ 1,\ 1,\ 0)$
$(\ 1,\ 0,\ 0,\ -1)$	$\equiv$	$\hat{w}$	$(\ 0,\ -1,\ -1)$
$(-1,\ 0,\ 1,\ 0)$	$=$	$\hat{t}+\hat{u}$	$(-1,\ 0,\ 1)$
$(\ 0,\ 1,\ 0,\ -1)$	$=$	$\hat{t}+\hat{w}$	$(-1,\ 0,\ -1)$
$(\ 0,\ -1,\ 0,\ 1)$	$=$	$\hat{u}+\hat{v}$	$(\ 1,\ 0,\ 1)$
$(\ 1,\ 0,\ -1,\ 0)$	$=$	$\hat{v}+\hat{w}$	$(\ 1,\ 0,\ -1)$
$(-1,\ 0,\ 0,\ 1)$	$=$	$\hat{t}+\hat{u}+\hat{v}$	$(\ 0,\ 1,\ 1)$
$(\ 0,\ 0,\ 1,\ -1)$	$=$	$\hat{t}+\hat{u}+\hat{w}$	$(-1,\ -1,\ 0)$
$(\ 0,\ 1,\ -1,\ 0)$	$=$	$\hat{t}+\hat{v}+\hat{w}$	$(\ 0,\ 1,\ -1)$
$(\ 1,\ -1,\ 0,\ 0)$	$=$	$\hat{u}+\hat{v}+\hat{w}$	$(\ 1,\ -1,\ 0)$

Table 6.1: The twelve vectors pointing to nearest neighbors of a FCC lattice, expressed in combinations of $\hat{t}$, $\hat{u}$, $\hat{v}$ and $\hat{w} = -(\hat{t}+\hat{u}+\hat{v})$.

6.3 Implementation of the extended repton model

While monomers in the projected repton model live on a one-dimensional lattice, the monomers in the model that we propose for polymer melts live on a FCC lattice. It is helpful to note that the three-dimensional hyperplane, located in a four-dimensional hypercubic space through the origin and normal to the vector $(1, 1, 1, 1)$, is such a FCC lattice. Stated differently, the set of points $\vec{x} = (a, b, c, d)$ with integer-valued coordinates, constrained to $a + b + c + d = 0$, forms a FCC lattice. The twelve vectors pointing to nearest-neighbor sites are $\hat{t} = (-1, 1, 0, 0)$, $\hat{u} = (0, -1, 1, 0)$, $\hat{v} = (0, 0, -1, 1)$, $\hat{w} = -(\hat{t} + \hat{u} + \hat{v}) = (1, 0, 0, -1)$, and some of their combinations, as listed in Table 6.1. The vector $(0, 0, 0, 0)$ is used as the representation for a zero-length bond.

As in the projected repton model, the state of a polymer on our FCC lattice can be specified by the location $\vec{x}_1$ of the first monomer, plus additionally the direction in which adjacent neighbors are located. We choose for the extended repton model to store the four directional bits in a single word: bits k, $k+16$, $k+32$ and $k+48$ of the 64-bit word D_i indicate the vector pointing from monomer i to monomer $i+1$. Thus, sixteen polymers are updated simultaneously.

The coordinates of the i^{th} monomer in polymer k can then be retrieved by summing over all words D_i bits k, $k + 16$, $k + 32$, and $k + 48$, yielding respectively the numbers n_t, n_u, n_v and n_w; the monomer position is then $\vec{x}_i = n_t\hat{t} + n_u\hat{u} + n_v\hat{v} + n_w\hat{w}$. Note that these summations require only $3i$ operations (a right shift of k bits, masking the direction bits, and adding to the sum), since the summation in the different bits can be done in a single operation. Since in the polymer melt model we need the coordinates frequently (every time we attempt a possible sideways move), and since our polymers are often several hundred monomers long, we do not keep track explicitly of only the position $\vec{x}_1$ of the first monomer, but also of the last monomer as well as a few other monomers along the chain, such that the distance along the chain to a monomer with known position is always less than 15. Of course, to retrieve the position $\vec{x}_i$, we start from the nearest monomer with known position in either direction along the polymer.

The implementation proceeds analogously to the projected repton model. Also here, if an interior monomer can move in one direction, its move in the other direction is blocked; this can be exploited as in the repton model, by combining two elementary moves. The precise operations are:

$$\begin{aligned} y_0 &= D_i \vee (D_i \gg 32) \\ y_1 &= y_0 \vee (y_0 \gg 16) \\ z_0 &= D_j \vee (D_j \gg 32) \\ z_1 &= z_0 \vee (z_0 \gg 16) \\ m_0 &= (y_1 \oplus z_1) \wedge (2^{16} - 1) \\ m_1 &= (D_i \oplus D_j) \wedge (M * m_0) \\ D'_i &= D_i \oplus m_1 \\ D'_j &= D_j \oplus m_1 \end{aligned} \tag{6.3}$$

Here, $A \gg k$ stands for shifting word A over k bits to the right, and $M = 2^0 + 2^{16} + 2^{32} + 2^{48}$ is a constant, used to duplicate the low 16 bits in the higher bits of the word. Thus, with only 15 operations, 2 loads, and 2 stores, we have performed 32 elementary moves. On a fast workstation, the above implementation requires 1.25 ns CPU time per elementary move.

As in the projected repton model, if monomers are displaced whose positions are tracked, these positions have to be updated in a loop over the 16 polymers. We succeeded in implementing those moves in 90 ns CPU time per elementary move. Displacement of the first and last monomers is attempted at twice the rate of the other monomers, for the same reason as in the projected repton model.

Besides of reptation moves, the dynamics consists of sideways moves. If a sideways move is attempted on monomer i, the relevant bit patterns are those indicating the direction from monomer $i-1$ to i and from i to $i+1$. These bit patterns, as listed in table 6.1, are all numbers in the range 0 to 15. The number 0 denotes stored length, and the numbers 5, 10, and 15 do not occur; the other 12 numbers denote bonds to the twelve nearest-neighbor sites. For every combination (D_i, D_j) of those bit patterns, we have pre-computed a list of all bit patterns D'_i and D'_j that may result after a sideways move. Depending on the combination (D_i, D_j), at most four different sideways moves can be proposed. In one step, we first select randomly the monomer number i and the list number k; then we attempt a sideways move of monomer i to the position determined by the k^{th} list; finally, if this move does not lead to overlapping monomers, it is accepted. This check for overlap requires computing the position of monomers, which requires computing the distance to the nearest tracked monomer. In our implementation, in which this distance is at most 15 monomers, the total CPU time required per such step equals 90 ns.

Each of these three mechanisms is implemented in a way that ensures detailed balance, and the combination of these three mechanisms assures a complete exploration of phase space, i.e., ergodicity holds. The combination of detailed balance and ergodicity guarantees that eventually, configurations are generated according to the Boltzmann distribution.

Not all moves are attempted with the same frequency. Since the long-time dynamics is determined primarily by reptation, as argued by De Gennes, the time scale is set such that reptation moves in the interior of the polymers are attempted with unit rate. Sideways moves which do not increase the amount of stored length are attempted with some rate r_s; the most natural choice for this rate would be unit rate once more, but because of the much higher computational cost for these moves, we often chose some value of $r_s < 1$. As discussed above, consistency with the density of stored length of the projected repton model demands that moves in which the amount of stored length is increased are attempted with a rate which is $2/z$ times lower than the opposite moves.

The most mobile monomers hop away, sideways, with a total rate of $2r_s$. Moves in which the first or the last monomer of a polymer joins its neighbor along the chain can be viewed partly as a reptation move, and partly as a sideways move. On these grounds, we have decided to attempt these moves with rate $1 + r_s$. Consistency with respect to the density of stored length requires that moves in which the first or last monomer leaves its neighbor along the chain in a specific direction are attempted with a rate of $2(1 + r_s)/z$.

In our implementation, reptation moves take 1.25 nanoseconds CPU time on average; end-monomer moves and sideways moves take 90 nanoseconds on average. Reptation moves and end-monomer moves are attempted at a 30 times higher rate than the sideways moves. On average, a unit of time takes about 6 nanoseconds per monomer.

6.4 Phase separation of a binary polymer mixture

We performed a simulation of the phase separation of a binary polymer mixture with polymer types A and B. The A and B polymers interact with a short-range repulsion, described by the Hamiltonian

$$H = J \sum_{\langle \vec{r}, \vec{r}' \rangle} \delta(\sigma_{\vec{r}}, A)\ \delta(\sigma_{\vec{r}'}, B) + \delta(\sigma_{\vec{r}}, B)\ \delta(\sigma_{\vec{r}'}, A), \tag{6.4}$$

where the summation runs over all pairs of nearest-neighbor sites, and $\sigma_{\vec{r}}$ is A, B, or 0 if site $\vec{r}$ is occupied by a polymer of type A, B or empty, respectively. The repulsion between A and B polymers provides the driving force for the phase separation.

To incorporate the change in total energy due to a move, in order to impose detailed balance in the simulation, we apply a Metropolis accept-reject procedure [56] for accepting a proposed sideways move. If the change in total energy ΔE is negative or zero, the proposed move is always accepted, otherwise the acceptance probability is equal to $P_{\text{accept}} = \exp(-\Delta E/k_B T)$, in which k_B is the Boltzmann constant and T the temperature.

We simulated a system containing a total of 46 080 polymers of length $L = 100$ on a lattice of $N = 13\,824\,000$ sites, at inverse temperature $\beta J = 0.1$. The system evolves in time through reptation moves, at unit rate, in combination with sideways moves at a rate of $r_s = 1/30$. Figure 6.2 shows two-dimensional slices of the three-dimensional system at times $t = 0$, $t = 4.1 \cdot 10^5$, and $t = 3.3 \cdot 10^6$ MC time units.

At various times, we determine the two-point distribution function

$$n_{AB}(\vec{r}) = \frac{\frac{1}{N} \sum_{\vec{r}'} \delta(\sigma_{\vec{r}'}, A)\ \delta(\sigma_{\vec{r}' + \vec{r}}, B)}{\left(\frac{1}{N} \sum_{\vec{r}'} \delta(\sigma_{\vec{r}'}, A)\right)\ \left(\frac{1}{N} \sum_{\vec{r}'} \delta(\sigma_{\vec{r}'}, B)\right)}. \tag{6.5}$$

From this function we determine the spherically-averaged radial correlation function, defined as $g(r) = 1 - \langle n(\vec{r}) \rangle$. This function is 1 at $r = 0$, then decreases, and eventually approaches 0 for large r. After some time, the conserved dynamics give rise to damped oscillations in $g(r)$, see figure 6.3. The frequency of these oscillations can be characterized by the shortest distance r_0 at which $g(r)$ equals zero. The typical domain size $d(t)$ may then be estimated as twice this distance.

Figure 6.4 shows the domain size $d(t)$ of the phase-separating mixture at different times, as a function of the cubic root of the simulation time in Monte Carlo sweeps. The straight line shows that the domain size grows as $d(t) \sim t^{1/3}$. This is to be expected in a system with conserved order dynamics and without hydrodynamics, as in “Model B” [67, 68].

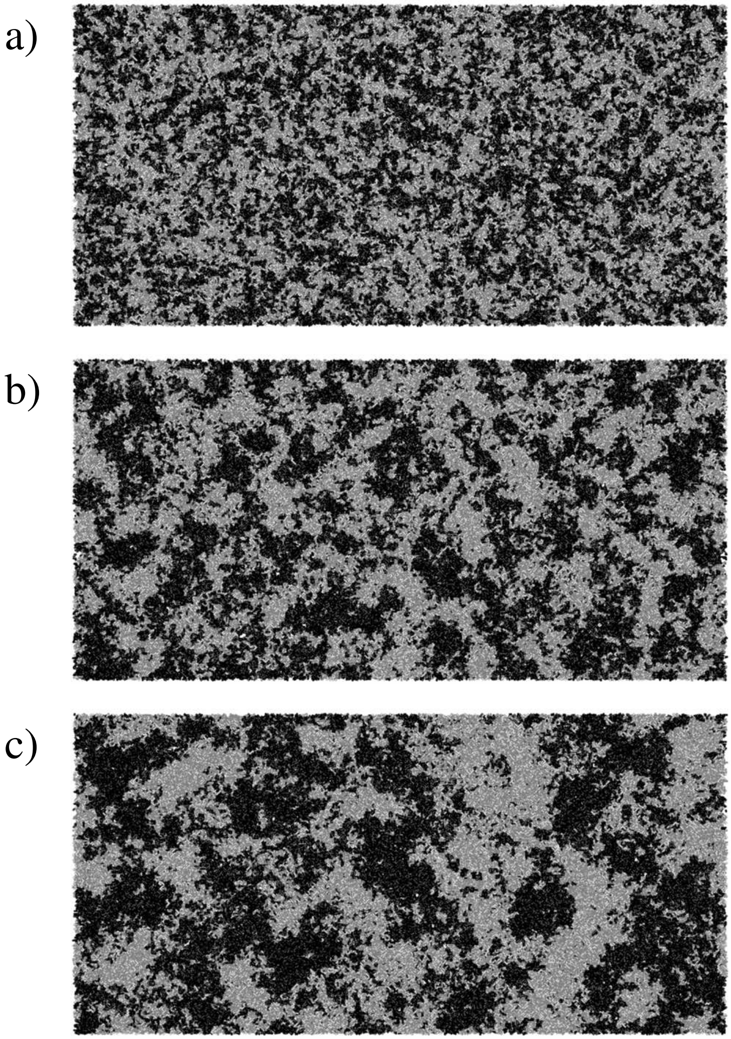

Figure 6.2: Two-dimensional snapshots of a three-dimensional simulation of a binary polymer mixture with 46 080 polymers of length $L = 100$ on a lattice of $N = 13\,824\,000$ sites, at inverse temperature $\beta J = 0.1$. Snapshot a) is taken at $t = 0$, i.e., the equilibrated system at infinite temperature taken as the initial configuration. Snapshots b) and c) are taken at $t = 4.1 \cdot 10^5$ and $3.3 \cdot 10^6$ MC time units.

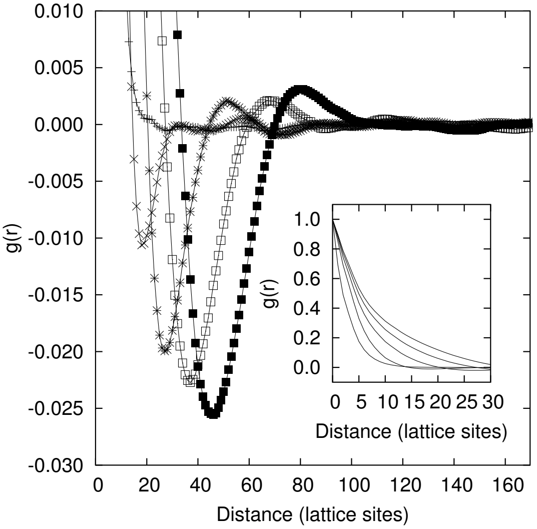

Figure 6.3: Two-point correlation function $g(r)$ of a phase-separating binary polymer mixture, at times $t = 0$, $5.2 \cdot 10^4$, $4.1 \cdot 10^5$, $1.4 \cdot 10^6$, and $3.3 \cdot 10^6$. At time $t = 0$, the temperature drops instantaneously from $T = \infty$ to below the phase-separation temperature. The correlation function shows a clear minimum, whose location moves to larger distances over time. The inset shows the same function at smaller distances.

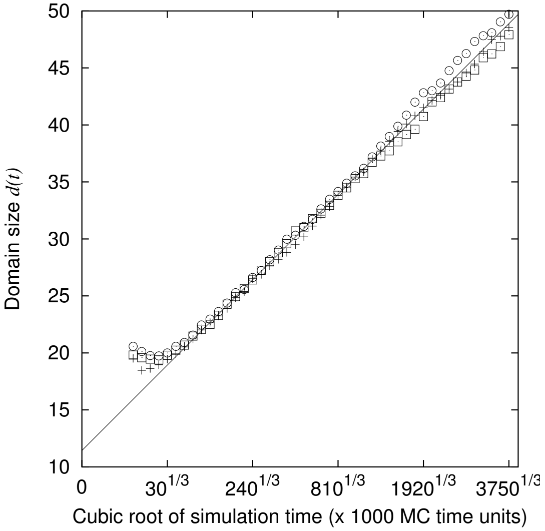

Figure 6.4: Domain size of a phase-separating binary polymer mixture as function of the cubic root of the simulation time. The three symbols indicate three independent simulations. A good agreement with domain growth $d(t) \sim t^{1/3}$ is found.

Chapter 7

Fractionation

If a solution of polydisperse A- and B-polymers undergoes phase separation, the molar mass distribution of the A-polymers in the A-rich phase becomes different from that in the B-rich phase. This phenomenon is known as fractionation.

Experiments on polymer mixtures have been reported by several groups [69–74]. These experiments investigate under which conditions phase separation sets in, and study dynamical properties of the phase-separation process. The properties of each of the separated phases have only been investigated by Edelman *et al.* [5], who have studied the composition of the separated phases of aqueous mixtures of gelatin and dextran. In section 7.1 we summarize the main findings of their paper. In section 7.2, we describe the equilibrium state of the system with Flory-Huggins theory. In section 7.3 we use computer simulations to compute the composition of the phases.

7.1 Experimental research

In experimental work, Edelman *et al.* [5] study the effect of phase separation on the molar mass distributions of aqueous mixtures of gelatin and dextran. These two biopolymers are both polydisperse. After full phase separation was established, samples of both phases were analyzed with size exclusion chromatography and multi angle laser light scattering (SEC-MALLS) to determine the molar mass distributions of both components in the two coexisting phases. Full phase separation of aqueous mixtures of gelatin and dextran results in a gelatin-rich and a dextran-rich phase. However, both phases contain either type of polymer [4]. Figure 7.1 shows the effect of phase separation on the molar mass distribution of gelatin (left) and dextran (right). It turns out that the minority type of molecules in each phase is dominated by the small molar mass part.

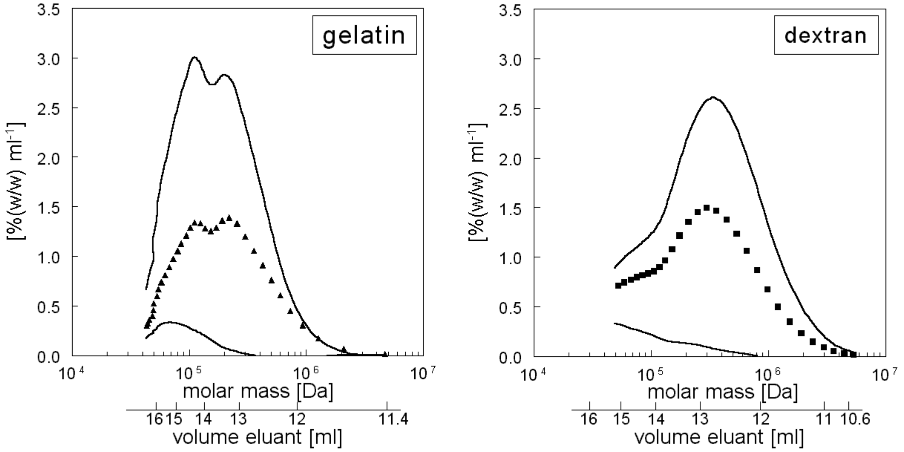

Figure 7.1: The effect of phase separation on the molar mass distributions of gelatin and dextran in coexisting phases after phase separation of a mixture containing 5% (w/w) gelatin and 5% (w/w) dextran, at 60°C. The triangles in the left figure and the squares in the right figure show the concentrations of gelatin (average molar weight $M_w = 184\,\mathrm{kDa}$) and dextran (average molar weight $M_w = 299\,\mathrm{kDa}$) in the mixture, before quenching below the phase separation temperature. The upper lines show the distributions in the rich phases and the lower lines the distributions in the poor phases.

To quantify this observation, the *degree of fractionation* is determined:

$$f^x(m) \equiv c^x_{\mathrm{poor}}(m)/c^x_{\mathrm{rich}}(m), \tag{7.1}$$

in which $c^x_{\mathrm{poor}}(m)$ and $c^x_{\mathrm{rich}}(m)$ are the concentrations of component x (gelatin or dextran) with a degree of polymerization m in the depleted (poor) and the enriched (rich) phases, respectively. The value $m = 1$ corresponds to a monomer. For gelatin, the monomer mass is taken 90 Dalton (the average mass of an amino acid in the gelatin used) and for dextran 162 Da (the mass of a glucose repeating unit).

If we plot $f^x(m)$ versus the degree of polymerization, it turns out that the fractionation is consistent with exponential behavior for both gelatin and dextran (see figure 7.2).

7.2 Flory-Huggins theory

In this section we apply Flory-Huggins theory to our particular lattice model of polydisperse polymer mixtures. First, we introduce the theory for a monodisperse symmetric mixture, and then extend it to general polydisperse systems.

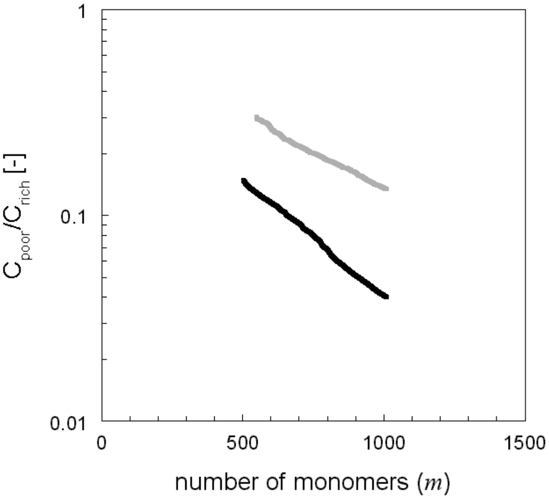

Figure 7.2: Degree of fractionation $f(m)$ as a function of polymer length (number of monomers), for a sample containing 5% (w/w) gelatin and 5% (w/w) dextran at T = 60°C. The grey and black lines correspond to gelatin and dextran, respectively.

The resulting expression for the free energy is minimized numerically for the molar mass distributions of the initial constituents of the polymer solutions used in the experiments and those used in our computer simulations.

For a binary polymer mixture with a repulsive interaction between the two types, phase separation is energetically favorable, but entropically unfavorable. The simplest theory of the phase transition in such systems originates from Flory and Huggins. Below, we present it first for a monodisperse symmetric mixture without the usual assumption that the system is large. The entropy of mixing is the logarithm of the number of ways one can split the set of polymers into a phase with p polymers of type A and q polymers of type B, and another phase with q polymers of type A and p polymers of type B. The entropy of mixing for either phase is

$$S = k_B \ln \frac{(p+q)!}{p! \cdot q!}, \tag{7.2}$$

which has a maximum at $p = q$, i.e., for two equal phases. Flory and Huggins were only interested in the limit where p and q are both very large, and used Stirling's approximation to arrive at the well-known result $S_{\mathrm{FH}} = -k_B p \ln \frac{p}{p+q} - k_B q \ln \frac{q}{p+q}$.

We assume that the polymers are randomly distributed throughout the system, and that effects of the chain structure on the energy of the system can be described by introducing an effective coordination number z_{eff}, instead of the lattice coordination number z. The total energy of the system, in this mean-field approximation, is proportional to the number of nearest-neighbor lattice sites occupied by different type monomers, with a proportionality constant J. The polymers consist of L monomers, and the system has V lattice sites. In a single phase, $(\rho_s/\rho_m)pL$ sites are occupied with one type of monomers, and $(\rho_s/\rho_m)qL$ sites are occupied with the other type, where ρ_s and ρ_m are the density of occupied lattice sites and the monomer density, respectively. Note that ρ_s and ρ_m are not equal, due to the possibility of several adjacent monomers occupying a single lattice site. With these assumptions, the polymers of type A have $(\rho_s/\rho_m)pLz_{\text{eff}}$ interactions with neighboring lattice sites, but only the fraction $(\rho_s/\rho_m)qL/V$ is occupied by monomers of type B. The energy is

$$E = \frac{\alpha J z}{V} L^2\, p\, q, \tag{7.3}$$

with $\alpha = (\rho_s/\rho_m)^2(z_{\text{eff}}/z)$.

The physical system in equilibrium, at constant pressure, will minimize the Gibbs free energy for a system, which is almost equivalent to minimizing the Helmholtz free energy, for constant volume, because the system is almost incompressible. The Flory-Huggins theory is based on the mean-field free energy expression for a system with a constant number of particles and a constant volume. The Helmholtz free energy of a single phase, $F = E - TS$, is given by

$$F = \frac{\alpha J z}{V} L^2\, p\, q - k_B T \ln \frac{(p+q)!}{p! \cdot q!}. \tag{7.4}$$

The free energy has a minimum at $H_p - c\,p = H_q - c\,q$, where $H_n = \sum_{i=1}^{n} i^{-1}$ is the n^{th} harmonic number, and $c = \alpha J z/V \cdot L^2/k_B T$ is the total effective interaction between the polymers. The harmonic numbers behave like $H_n \approx \ln n$ for large n, and the continuous version goes to zero for small n. Indeed, if we may assume large numbers of polymers for both polymer types, we can use $H_n \approx \ln n + \gamma$, and we arrive at the equation $\ln p - c\,p = \ln q - c\,q$, which is precisely the same result one would obtain by using the Flory-Huggins free energy instead of equation (7.2). The constant $\gamma = 0.577216...$ is known as Euler's constant.

A general phase-separated polydisperse polymer mixture can be described as a set of boxes $\{B\}$, containing polymers from a set of different species (types of polymers) $\{S\}$, and a set of different lengths $\{L\}$. The expression for the entropy becomes

$$S = k_B \ln \prod_{s\in\{S\}} \prod_{l\in\{L\}} \frac{(\sum_{b\in\{B\}} n_{slb})!}{\prod_{b\in\{B\}} n_{slb}!}, \tag{7.5}$$

where n_{slb} is the number of polymers of species s and length l present in box b. Maximum entropy is reached if all boxes have equal volume and contain the same amount of polymers of each type and length. We can rewrite the logarithm of the product as a sum of logarithms, and, in the limit where all $n_{slb} \gg 1$, use Stirling's approximation to avoid the factorials:

$$S_{\mathrm{FH}} = -k_B \sum_{s\in\{S\}} \sum_{l\in\{L\}} \sum_{b\in\{B\}} n_{slb} \ln n_{slb} + \text{const.} \tag{7.6}$$

The energy of mixing is again described by a mean-field approximation:

$$E = \frac{\alpha z}{2} \sum_{b\in\{B\}} \frac{1}{V_b} \sum_{l\in\{L\}} \sum_{l'\in\{L\}} \sum_{s\in\{S\}} \sum_{s'\in\{S\}} \epsilon'_{ss'}\; n_{slb}l \cdot n_{s'l'b}l' + \text{const.}, \tag{7.7}$$

where V_b is the number of lattice sites in box b, and $\epsilon'_{ss'} = \epsilon_{ss'} - \epsilon_{0s} + \epsilon_{00}$ is the effective interaction between particles s and s'.

We now turn to the polydisperse case with two components A and B, with only interactions between nearest-neighbor sites occupied by different types, i.e. $\epsilon_{AB} = J \neq 0$. It is convenient to change notation: we denote the number of A-polymers of length L_i in box b with p_i^b, and the number of B-polymers of length L_i in box b with q_i^b.The free energy then becomes:

$$\begin{aligned} F \;=\; & \sum_b \frac{\alpha J z}{V_b} \Big(\sum_i p_i^b L_i\Big)\Big(\sum_i q_i^b L_i\Big) + \\ & k_B T \sum_{b,i} p_i^b \ln p_i^b + k_B T \sum_{b,i} q_i^b \ln q_i^b + \text{const.} \end{aligned} \tag{7.8}$$

Again, V_b is the volume of box b. Given a starting distribution, we can minimize this free energy under the constraints that the total number of A- and B-polymers of each length is constant and that the total volume $V = V_1 + V_2$ is constant. In a symmetric binary mixture, the free energy of the two phases is equal, and $p_i^1 = q_i^2$ and $q_i^1 = p_i^2$. We can find the minimum of the free energy in one of the boxes explicitly by differentiating with respect to p_j for some j:

$$\frac{1}{k_B T}\frac{\partial F}{\partial p_j} = \frac{\alpha J z}{V_b k_B T} L_j \sum_i (p_i + q_i) L_i + \ln \frac{p_j}{q_j}. \tag{7.9}$$

This clearly shows that the fractionation of a symmetric binary polymer mixture is exponential in the polymer length.

We approximated the experimental length distribution $\rho(m)$ of polymers containing m monomers from the experimental molar mass distribution as follows. First, we expressed the experimental molar mass distributions (figure 7.1) in terms of polymer length. We have taken the average mass of a gelatin monomer as 90 Da (i.e., 90 grams/mol) and the mass of a dextran monomer as

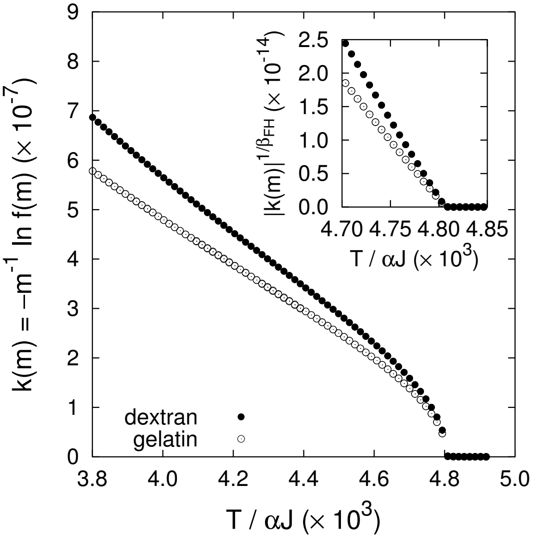

Figure 7.3: Flory-Huggins results for $k(m) \equiv -m^{-1} \ln f(m)$ as a function of dimensionless temperature, for dextran (upper points) and gelatin (lower points). The curves for all polymer lengths m coincide, showing that the degree of fractionation $f(m) = \exp(-k(m)m)$ increases exponentially with length. The inset shows that $|k(m)|^{1/\beta_{\mathrm{FH}}}$ decreases linearly with temperature, up to its critical value.

162 Da. These length distributions were then fitted by sums of gaussians with mean value $\ln(a_i)$ and width σ_i (four gaussians for gelatin, three for dextran):

$$\rho(m) \approx \sum_i k_i \exp\Big(-\frac{[\ln(m) - \ln(a_i)]^2}{2\sigma_i^2}\Big). \tag{7.10}$$

Next, the sum of these approximated length distributions for gelatin and dextran was divided into 100 bins containing the same number of polymers. Each bin is then represented by a monodisperse population with the average length, and appropriately chosen fractions of dextran and gelatin.

Using the Flory-Huggins expression, we minimized the free energy for a configuration in two boxes. We expect that the degree of fractionation is exponential in the length of the polymers, i.e., $f(m) \sim \exp(-km)$, and therefore plot

$$k(m) \equiv -m^{-1} \ln f(m) \tag{7.11}$$

as a function of temperature, see figure 7.3. As expected, measurements for different polymer lengths fall on top of each other in this figure, showing that the degree of fractionation $f(m)$ changes exponentially with the polymer length.

The figure shows that the two boxes contain equal phases above $T/\alpha J = 4.80{\cdot}10^3$ and different phases below that value, one rich in gelatin, the other rich in dextran. In mean-field theory, we expect that the critical exponent for phase separation is $\beta_{\text{FH}} = 1/2$. In the inset of the figure we plot $|k(m)|^{1/\beta_{\text{FH}}}$ in the critical regime, which indeed shows linear behavior up to the critical temperature.

7.3 Computer simulations

We performed computer simulations of the model, with polydispersity as described above, containing 5 400 polymers in total, on a FCC lattice with 1 728 000 sites. Since we are interested in equilibrium properties of the simulated polymer mixture, and not in the exact dynamics that leads to this equilibrium, we are not limited to using mechanisms that reflect realistic dynamics of polymer mixtures. In fact, we used an additional mechanism which is highly artificial. In this mechanism, the proposed move is to change the type of a randomly chosen single polymer from A to B or vice versa. This is then accepted or rejected, according to the Metropolis algorithm. This mechanism does not alter equilibrium properties, but should not be used if dynamical properties are studied.

Our computer simulations are limited to identical polydispersity for the two types of polymers. We fitted a single gaussian to the average of the length distributions of gelatin and dextran, and then proceeded as in section 7.2 to obtain a population of 45 different lengths. The average polymer length within the simulations was 165 monomers. The average molar mass of this distribution was $M_w = 214$ monomers, as compared to a few thousands in the experiments. All simulations start with a system generated and equilibrated without nearest-neighbor interactions, i.e., at infinite temperature. We bring the system in equilibrium at the selected temperature. We keep track of the number of polymers of type A and B for each polymer length. The averages of these data are used to determine the phase separation temperature and the fractionation of the different polymer lengths.

With decreasing temperature, the binary polymer mixture separates into two equivalent phases. The familiar three-dimensional Ising model also phase-separates into two phases due to short-range interaction, so critical behavior equal to the universality class of the three-dimensional Ising model is expected. A subtle issue is that in our simulations the molar mass distribution is kept constant; this could give rise to so-called ‘hidden variables’ in the context of Fisher renormalization [75, 76]. If this is indeed the case, then the specific heat will take large values, but does not diverge to infinity at the critical point. Consequently, the corresponding critical exponent α_c' is zero, and the other exponents take slightly different values; in particular $\beta' = \beta/(1 - \alpha_c) \approx 0.37$,

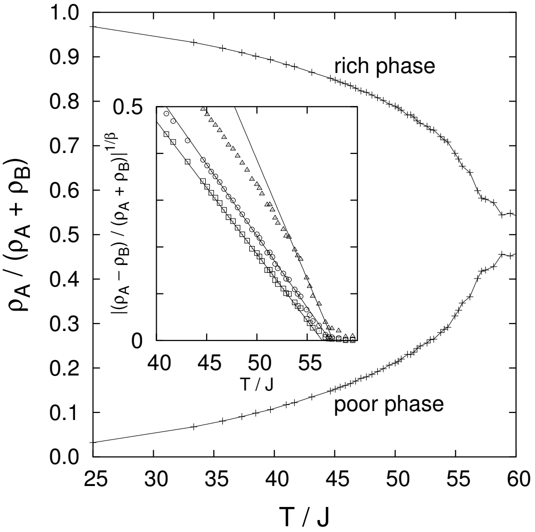

Figure 7.4: Relative weight $\rho_A/(\rho_A + \rho_B)$ of A-polymers in their rich (upper curve) and poor phase (lower curve), as a function of temperature. The inset shows $|(\rho_A - \rho_B)/(\rho_A + \rho_B)|^{1/\beta}$, which is expected to decrease linearly to zero at the critical temperature. We plotted the data with three values for the exponent: $\beta = 0.3269$ (squares) as in the 3D Ising model, the Fisher-renormalized value $\beta = 0.37$ (circles), and the mean-field value $\beta = 0.5$ (triangles). The straight lines are fits by eye, fitting the critical temperature (zero-crossing) and amplitude (slope). Our data is well described with $\beta = 0.3269$ or $\beta = 0.37$, but is incompatible with the mean-field value $\beta = 0.5$. Above the critical temperature, the rich phase and poor phase are still distinguishable due to the finite size of the box.

instead of $\beta = 0.3269$ in the pure Ising model [77].

In the Ising model, if the critical temperature is approached from below, the magnetization M decreases to zero as $M \sim (T_c - T)^\beta$. In our model, the equivalent of the magnetization is the normalized density difference $|(\rho_A - \rho_B)/(\rho_A + \rho_B)|$. This quantity, raised to the power $1/\beta$, is thus expected to decrease linearly to zero at the critical temperature. The inset of figure 7.4 shows this approach for $\beta = 0.3269$ as in the Ising model, $\beta = 0.37$ as expected if Fisher renormalization occurs, and the mean-field value $\beta = 0.5$. Mean-field behavior is inconsistent with our data, but we cannot establish whether Fisher renormalization takes place, within our numerical accuracy.

The degree of fractionation of the polymers in the computer simulations shows exponential dependence on the polymer length for long polymers in the simulation. The short polymers deviate slightly due to end group effects: a poly-

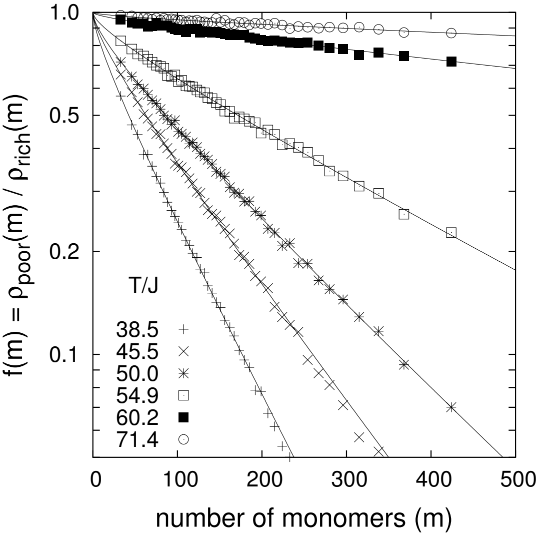

Figure 7.5: Degree of fractionation $f(m)$, as a function of polymer length, measured in the number of monomers m, for temperatures T/J = 38.5, 45.5, 50.0, 54.9, 60.2, and 71.4. Lines are fits to the function $f'(m) = \exp(-km - k_2\sqrt{m})$.

mer of length $m > 1$ has less exposed length than m times that of one monomer. This finite-length effect can be accounted for with $f'(m) \sim \exp(-km - k_2\sqrt{m})$. In figure 7.5 the measured values for $f(m)$ are plotted as a function of m. In the same figure, curves for $f'(m)$ with fitted values for k and k_2 are shown as well. The long-polymer limit is retrieved by setting k_2 to zero; in this limit, $k = -m^{-1} \ln f(m)$ is independent of polymer length m. Figure 7.6 shows the values of k as a function of temperature.

We also applied Flory-Huggins theory for the length distribution used in the simulations. Figure 7.7 shows the resulting $k(m) = -m^{-1} \ln f(m)$ as a function of temperature. Well below the critical temperature, the short polymers deviate from the relation $f(m) \sim \exp(-km)$, toward a higher degree of fractionation. This deviation decreases with increasing polymer lengths and disappears completely if Stirling's approximation is used.

A comparison of figure 7.7 with figure 7.6 shows quantitative agreement well below the critical temperature if $\alpha = 0.521$. The value of α describes the combined effect of the stored length of the polymers, which decreases the number of interactions, and the effective coordination number z_{eff}, which is smaller than the lattice coordination number z because of shielding of monomers due to neighbors along the chain. Both contributions are insensitive to small temper-

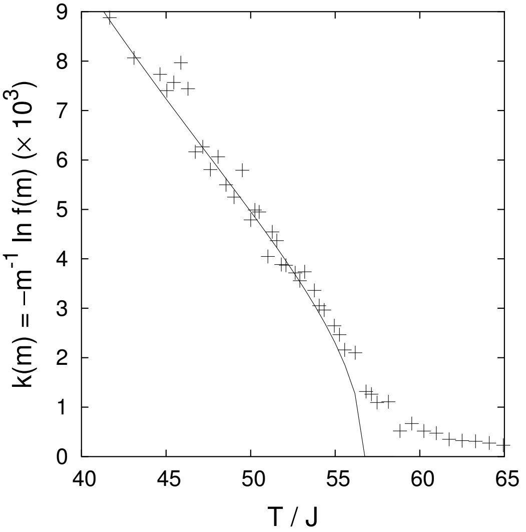

Figure 7.6: Numerical values for the fitting parameter k in the function $f'(m) = \exp(-km - k_2\sqrt{m})$ fitted as in figure 7.5, as a function of dimensionless temperature. The line in the fit is as shown in figure 7.7, after substitution of $\alpha = 0.521$.

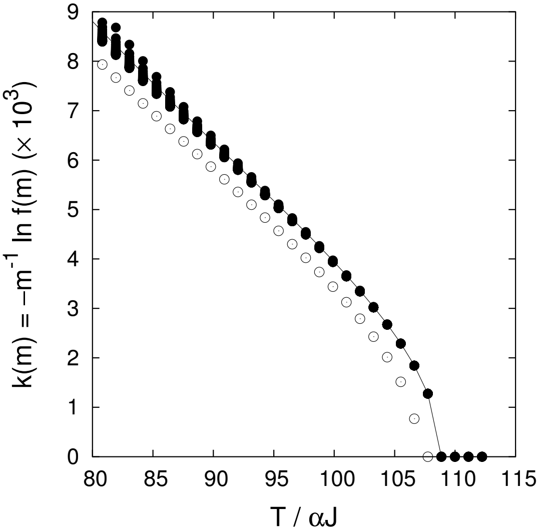

Figure 7.7: Flory-Huggins results for $k(m) = -m^{-1} \ln f(m)$ as a function of dimensionless temperature, for the polymer length distributions used in the computer simulations, with and without Stirling's approximation (lower and upper points, respectively).

ature changes. Close to the critical temperature we do not expect quantitative agreement, due to the finite simulation cell.

7.4 Comparisons

In summary, we discussed the experimental result that fractionation is an exponential function of the molar mass: $f(m) \sim \exp(-km)$. We showed that this scaling is supported by Flory-Huggins theory and computer simulations. Secondly, we studied the exponent β which describes the behavior of phase separation as a function of temperature close to its critical value. Within Flory-Huggins theory, this exponent was found to be $\beta_{\mathrm{FH}} = 0.5$, consistent with the mean-field estimation of the exponent β in the three-dimensional Ising model. In computer simulations, on the other hand, this exponent was found to be consistent with non-mean-field values $\beta = 0.3269$ as found in simulations of the three-dimensional Ising model, as well as with $\beta = 0.37$, the Fisher-renormalized critical exponent.

Appendix A

Parallel matrix-vector multiplication

The reduced transition matrix of the three-dimensional cage model for gel electrophoresis, described in section 3.2, becomes excessively large for polymer lengths more than $L = 12$. Parallel machines often have more memory than commonly used sequential machines such as workstations or PCs and this memory can be used to solve larger problems. Our task is then to distribute the matrix over the processors, such that the problem can be solved as efficiently as possible, hopefully also improving the performance by a factor close to the number of processors used.

A.1 BSP

A bulk synchronous parallel (BSP) program operates by alternating between a phase where all processors simultaneously compute local results and a phase where they communicate with each other. A superstep in a BSP algorithm consists of a computation phase followed by a communication phase. Before and after each communication phase a global synchronization is carried out. The BSPlib library (for the programming language C) [78, 79] consists of only 20 primitives and is based on one-sided communications. One-sided communications, as opposed to two-sided communications, cannot create deadlock situations. The communication mechanisms built into the BSP library are *remote write*, *remote read* and *bulk synchronous message passing.* In all three cases the remote processor is, at least conceptually, passive in the current superstep. The basic communication primitives are summarized below.

- Remote write: the processor that executes a **put** statement copies a block of memory to a remote memory address at the time of the next synchronization.

- Remote read: the processor that executes a **get** statement copies a block of memory from a remote memory address at the time of the next synchronization.

- Bulk synchronous message passing: the processor that executes a **send** statement sends a message, consisting of a tag and a payload part, to the buffer of a remote processor at the time of the next synchronization. The messages can be read from the buffer by a **move** operation after the next synchronization.

The BSP cost model consists of four parameters: the number of processors p, the speed of the processors s, the communication time g and the synchronization time l. The speed of the processors is measured as the number of floating point operations per second. The communication time is measured as the average time taken to communicate a single word to a remote processor, when all the processors are simultaneously communicating; the unit of time is the time per floating point operation (flop). The synchronization time is the amount of time needed for all processors to synchronize, also measured in flop time.

As mentioned earlier a BSP program is either in a computing phase or in a communication phase. This makes predicting the performance of algorithms much easier than in the case of parallel programming models where computation and communication are interleaved in a less structured fashion. The analysis of the cost of a superstep is relatively simple. For each processor i we count the number of flops w_i, the number of words sent to other processors $h_i^{(s)}$ and the number of words received $h_i^{(r)}$. The time taken by processor i for computation is w_i and for communication is $h_i = \text{Max}(h_i^{(s)}, h_i^{(r)})$. The cost of the superstep is $\text{Max}_i(w_i) + \text{Max}_i(h_i)g + l$. This shows that optimally we should divide the problem to be solved in equal parts, in the sense that the calculations and communications are evenly distributed over the available processors. Of course, we should also take care to reduce the total amount of communication.

A.2 Matrix distribution

A good way to distribute an $n \times n$ dense matrix over $p = MN$ processors is a generalized $M \times N$ block/cyclic distribution: the rows are divided into p row blocks of equal size and the columns into N column blocks of equal size; then

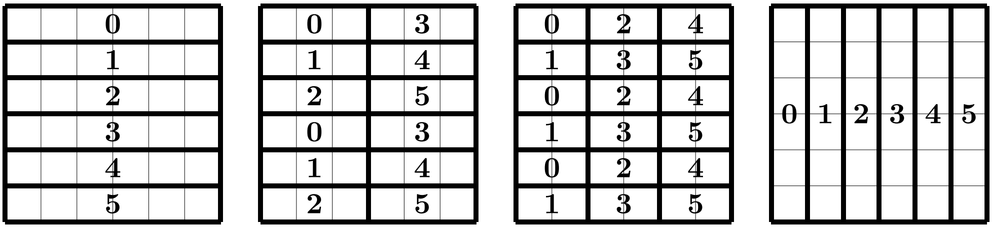

Figure A.1: $M \times N$ generalized block/cyclic distribution for matrices on $p = MN = 6$ processors. The rows have a block-cyclic distribution, with p blocks which are cyclicly numbered $0, 1, \ldots, M-1, 0, 1, \ldots$, and the columns have a block distribution, N blocks numbered $0, 1, \ldots N-1$. From left to right: $M = 6, N = 1$; $M = 3, N = 2$; $M = 2, N = 3$ and $M = 1, N = 6$.

the matrix elements a_{ij} are assigned to the processors as follows:

$$\begin{aligned} \phi_0(i) &= (i \operatorname{div} \tfrac{n}{p}) \bmod M, \\ \phi_1(j) &= j \operatorname{div} \tfrac{n}{N}, \\ a_{ij} &\longmapsto P(\phi_0(i) + M\phi_1(j)), \end{aligned} \tag{A.1}$$

as shown in figure A.1. The vector elements are best distributed to the same processor as the diagonal of the matrix. Note that for each generalized block/cyclic distribution: all processors have an equally large part of the matrix; each column is distributed over M processors; each row is distributed over N processors; each processor has the same number of submatrices and each processor has the same number of diagonal elements. This scheme fits within the general Cartesian framework of the work of Bisseling and McColl [80]; it is similar but not identical to the block/cyclic distribution.

The approach of Bisseling and McColl to the matrix vector product $\vec{r} = A\vec{x}$ can be divided into four stages:

- fan-out: the elements x_j are communicated to the processors containing the values a_{ij};
- local matrix-vector multiplications: the partial results $u_{it} = \sum_j' a_{ij} x_j$ are computed, with the sum taken over only the local values of a_{ij}, which all have the same $t = \phi_1(j)$;
- fan-in: the partial results, $u_{i\phi_1(j)}$, of the processors are sent to the processor that possesses the corresponding element r_i;
- summation of the partial results: $r_i = \sum_{t=0}^{N-1} u_{it}$.

If the matrix is divided into rows (which is the special case $N = 1$ for our generalized block/cyclic distribution), the fan-in and summation of partial sums is avoided; this saves some communication, but all processors then have to communicate with all other processors in the fan-out part. On the other hand, if the matrix is divided into columns ($M = 1$), then the fan-out communication is avoided and the fan-in communication is an all-to-all operation. For the general $M \times N$ distribution, the fan-out is an M-to-M communication and the fan-in an N-to-N communication. The communication then takes $\mathcal{O}((M + N)\frac{n}{p})g$ time, instead of $\mathcal{O}(M \cdot N \cdot \frac{n}{p})g$. The communication is minimal if $M = N = \sqrt{p}$ is used.

For a sparse matrix, the algorithm is adapted to avoid computations and communications involving zero elements: elements x_j are only sent if the corresponding $a_{ij} \neq 0$; partial sums are only computed using products $a_{ij}x_j$ with $a_{ij} \neq 0$ and the partial sums are only sent and summed if they are nonzero. The next section shows how advantage is taken of the specific sparsity structure of the matrix.

A.3 Exploiting the sparsity structure

In our problem, for $L > 12$, we cannot afford to store the complete matrix on a single processor, so we need to distribute it over a number of processors. The matrix we have to deal with is sparse and we exploit this in our computations, since we only handle nonzero elements A_{ij}. The standard approach to communicate a subset of elements of a vector is to gather all elements and their global indices in separate arrays, and then sending those arrays to the processors that need them. The overhead of repeatedly sending the same arrays with indices may be removed by sending them only the first time the matrix vector multiplication is performed, but the overhead of repeatedly packing and unpacking the vector elements cannot be removed in general.

Our transition matrix has a particular structure with “patches” with many nonzero elements. We exploit this to make communications faster by sending contiguous subvectors, avoiding the packing and unpacking overhead. Consider a rectangular patch (i.e., a contiguous submatrix). A value x_j must be sent to the owner of the patch if an element A_{ij} in column j of the patch is nonzero. It is likely that most columns of the patch have at least one nonzero, so we might as well send all x_j for that patch. This makes it possible to send a contiguous subvector of $\vec{x}$, which is more efficient than sending separate components; this comes at the expense of a few unnecessary communications. The trade-off can be shifted by increasing or decreasing the patch size.

To find suitable patches, we first divide the state vector into contiguous

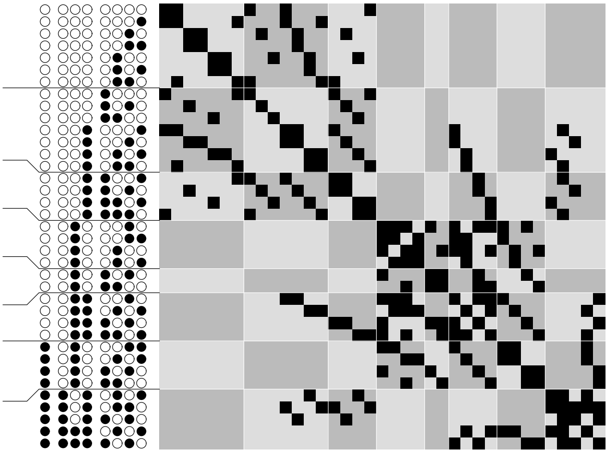

Figure A.2: Reduced transition matrix for polymer length $L = 5$. The size of the matrix is 37×37 and it has 233 nonzero elements, shown as black squares. To the left of each row is the corresponding kink representation written as a binary number, with black circles denoting 1 and open ones 0. The horizontal lines on the left show the initial division of the reduced state vector into eight contiguous parts, optimized to balance the number of nonzeros in the corresponding matrix rows. The jumps of these lines indicate slight adjustments to make the division fit the nonzero structure of the matrix. The resulting vector division induces a division of the rows and columns of the matrix, and hence a partitioning into 64 submatrices, shown by the gray checkerboard pattern. Complete submatrices are now assigned to the processors of a parallel computer.

subvectors. We use a heuristic to partition the matrix into blocks of rows with approximately the same number of nonzeros. If we use P processors, and we want each processor to have K subvectors, we have to divide the vector into KP subvectors. (The factor K is the overpartitioning factor.) This initial division tries to minimize the computation time. Next, we adjust the divisions to reduce communication: a suitable patch in the matrix corresponds to an input subvector of kink representations where only the last few bits differ, and also to an output subvector with that property. Therefore, we search for a pair of adjacent kink representations that has a different bit as much as possible to the left. This is a suitable place to split. We try to keep the distance from the starting point as small as possible.

As an example of the structure of the reduced transition matrices and the division into submatrices, we show the nonzero structure of the matrix for $L = 5$ in figure A.2 and its corresponding communication matrix in figure A.3 (left). The communication matrix is built from the partitioned transition matrix, by considering each submatrix as a single element. It is a sparse matrix of much

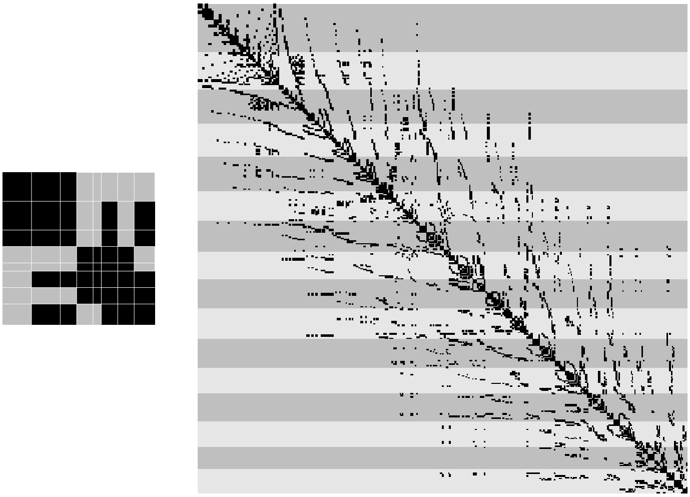

Figure A.3: Communication matrix for $L = 5$ (left) and $L = 13$ (right). Note that the matrix for $L = 5$ can be obtained by replacing each nonempty submatrix in Fig. A.2 by a single nonzero element. The communication matrix for $L = 13$, of size 320×320, is distributed over 16 processors in a row distribution.

smaller size which determines the communication requirements. Our communication matrix for $L = 13$ is given in figure A.3 (right).

A.4 Timings

Our computations were performed on a Cray T3E computer. The peak performance of a single node of the Cray T3E is 600 Mflop/s for computations. The bsp_probe benchmark shows a performance of 47 Mflop/s per node [78]. The peak interprocessor bandwidth is 500 Mbyte/s (bidirectional). The bsp_probe benchmark shows a sustained bidirectional performance of 94 Mbyte/s per processor when all 64 processors communicate at the same time. This is equivalent to a BSP parameter $g = 3.8$, where g is the cost in flop time units of one 64-bit word leaving or entering a processor. The measured global synchronization time for 64 processors is 48 μs, which is equivalent to $l = 2\,259$ flop time units.

Table A.1 presents the execution time of one iteration of the algorithm in two forms: the BSP cost $a + bg + cl$ counts the flops and the communications and thus gives the time on an arbitrary computer with BSP parameters g and l, whereas the time in milliseconds gives the measured time on this particular architecture, split into computation and communication time. (The total mea-

L	P	BSP cost	time (ms)	efficiency	speedup
12	8	$545\,156 + 64\,716g + 2l$	47 + 4.3	85%	6.8
13	16	$1\,002\,824 + 187\,347g + 2l$	89 + 13	81%	13.0
14	32	$1\,836\,920 + 425\,152g + 2l$	169 + 44	73%	23.4
15	64	$3\,452\,776 + 1\,380\,415g + 2l$	330 + 112	67%	42.9

Table A.1: BSP cost, time, efficiency, and speedup for one matrix-vector multiplication.

sured synchronization time is negligible.) The BSP cost can be used to predict the run time of our algorithm on different architectures. Table A.1 also gives the efficiency and speedup relative to a sequential program.

Peak computation performance is often only reached for dense matrix-matrix multiplication; the performance for sparse matrix-vector multiplication is always much lower. Comparing the flop count and the measured computation time for the largest problem $L = 15$, we see that we achieve about 10.5 Mflop/s per processor. Comparing the communication count with the measured communication time, we obtain a g-value of 8.1 μs, (or $g = 3.8$ flop units; see above). This means that we attain the maximum sustainable communication speed. This is due to the design of our algorithm, which communicates contiguous subvectors instead of single components. Furthermore, the results show that our choice to optimize mainly the computation (by choosing a row distribution) is justified for this architecture: the communication time is always less than a third of the total time. For a different machine, with a higher value of g, more emphasis must be placed on optimizing the communication, leading to a two-dimensional distribution.

Each iteration of our computation contains one matrix-vector multiplication. The number of iterations needed for convergence depends on the length of the polymer, and on the applied electric field. The iteration was stopped when either the accuracy was better than 10^{-10}, or the number of iterations exceeded 100 000. In the latter case, the accuracy was computed at termination. Typically, for $L = 15$ and a low electric field strength, 50 000 iterations are needed, taking about 6 hours per data point. Only computed values with accuracy 10^{-4} or better are shown in figure 5.3. For $L = 12$, we compared the output for the parallel program with that of the sequential program and found the difference to be within rounding errors. The total speedup for $L = 15$, compared to a naive implementation (for which one would need 38.5 Tbyte of memory), is a factor 1.5×10^6: a factor of 17 248 by using a reduced state space, a factor of 2 by shifting the eigenvalues of the reduced transition matrix, and a factor 42.9 by using a parallel program on 64 processors.

Appendix B

Proof of correctness of the kink-representation approach

Our aim is to prove that all polymer configurations with the same kink representation, as discussed in section 5.3, have the same probability in the steady state. I thank Rob Bisseling for providing this proof, which has been published as part of reference [3]. To prove the correctness of the kink-representation approach, it is sufficient to show that two configurations with the same kink representation can move to the same set of six kink representations with the moving of a certain kink or end monomer. We prove this by giving a procedure for determining the resulting six kink representations.

First, we introduce our notation. Define $R(i, j)$ as the statement "the part of the configuration between monomers i and j is removable", where $0 \leq i, j < L$. (By this definition, $R(i, i)$ holds.) Define $S(i, j)$ as "monomers i and j are at the same site". Define $\text{sign}(i) = 1$ if bond $[i, i + 1]$ is in the direction of the electric field, and $\text{sign}(i) = -1$ otherwise. We have the following useful properties.

1. $R(i, j)$ implies $S(i, j)$ and $j - i$ even.

2. Let $i < j < k$. If $R(i, k)$ and j is the center of a kink, then the part between i and k can be removed starting with the kink at j. Proof: by induction on the length of the part.

3. The relations R and S are equivalence relations between monomers, i.e., they are reflexive, symmetric, and transitive. Proof: trivial, except for the proof of the transitivity of R, which uses the previous property. For example, let $i < j < k$. If $R(i, k)$ and $R(i, j)$, then a removal of the part $[j, k]$ can be obtained by starting the removal of $R(i, k)$ by removing kinks in $[i, j]$.

4. Let j be the smallest integer such that $j > i$ and $R(i, j)$. Then $R(i+1, j-1)$. Proof: by induction on the length of $[i, j]$.

5. Let i, i', j, j' be monomers with $|i-i'| = |j-j'| = 1$. If $R(i, j)$ and $S(i', j')$, then $R(i', j')$. Proof: we treat the case $i' = i + 1$ and $j' = j + 1$ as an example. First, we extend the part $[i, j']$ with a dummy monomer $i - 1$ at the site of i'. We can remove $[i - 1, j']$ by first removing $[i, j]$ and then removing the remaining kink $[i - 1, j']$. By Property 2 above, we can also start with kink $[i - 1, i']$ and then remove $[i', j']$. Hence $R(i', j')$.

Now assume that the kink at i of a given polymer configuration moves. (Moves of end monomers can be treated similarly.) We present a procedure for generating the resulting six kink representations, which is based solely on the original kink representation, i.e., on the relation R and the bond signs. The correctness proof of this procedure uses the properties above; for brevity, we omit the details. A kink exists at i if and only if $R(i - 1, i + 1)$. In that case, $\mathrm{sign}(i) = -\mathrm{sign}(i - 1)$. The set of removable parts $[x, y]$ with $x, y \neq i$ does not change; changes can only occur if $x = i$ or $y = i$. The procedure checks for all j whether $R(i + 1, j)$. If so, monomer i can move to the sites of monomers $j - 1$ and $j + 1$, provided these monomers exist. This is because $j - 1$, $j + 1$, and i are all at distance one from the site of j. If $j - 1 = i$, then $R(i + 1, j)$ holds, and the move to $j - 1$ is the identity move, which does not change the kink representation. Assume the move is to $j - 1$ (the case $j + 1$ is similar). Assume $j - 1 \neq i$. The new set of $x \neq i$ with $R(i, x)$ equals the old set of $x \neq i$ with $R(j - 1, x)$. The new $\mathrm{sign}(i)$ equals the old $\mathrm{sign}(j - 1)$.

The generated moves are collected and duplicates are removed by using the old relation R. For example, if $R(i + 1, j)$ and $R(i + 1, j')$ and we have to check whether moves to $j - 1 \neq i$ and $j' - 1 \neq i$ are identical, i.e., whether $S(j - 1, j' - 1)$, we can do this by checking the old $R(j - 1, j' - 1)$. The total number of moves after duplicate removal is at most six. To make the total six, extra moves are added. This is done such that three moves have $\mathrm{sign}(i) = 1$ and the others $\mathrm{sign}(i) = -1$. The relation R after such an extra move is the same as before the moves, except that $R(i, x)$ becomes false for all $x \neq i$. Note that $R(i, x)$ with $x > i$ implies that there exists a smallest $x' > i$ with $R(i, x')$, and this in turn implies $R(i + 1, x' - 1)$, so that the corresponding move of i to x' must have been generated previously.

Bibliography

[1] A. van Heukelum and G. T. Barkema. Lattice models of DNA electrophoresis. *Electrophoresis*, 23 (16):2562–2568, 2002.

[2] A. van Heukelum and H. R. W. Beljaars. Electrophoresis simulated with the cage model for reptation. *J. Chem. Phys.*, 113 (9):3909–3915, 2000.

[3] A. van Heukelum, G. T. Barkema, and R. H. Bisseling. DNA electrophoresis studied with the cage model. *J. Comput. Phys.*, 180 (1):313–326, 2002.

[4] M. W. Edelman, E. van der Linden, E. H. A. de Hoog, and R. H. Tromp. Compatibility of gelatin and dextran in aqueous solution. *Biomacromolecules*, 2 (4):1148–1154, 2001.

[5] M. W. Edelman, R. H. Tromp, and E. van der Linden. Phase separation induced fractionation in molar mass in aqueous mixtures of gelatin and dextran. *Phys. Rev. E*, 67 (2):021404, 2003.

[6] A. van Heukelum and G. T. Barkema. Reaching large lengths and long times in polymer dynamics simulations. *J. Chem. Phys.*, 119 (15):8197–8202, 2003.

[7] A. van Heukelum, G. T. Barkema, M. W. Edelman, E. van der Linden, E. H. A. de Hoog, and R. H. Tromp. Fractionation in a phase-separated polydisperse polymer mixture. *Macromolecules*, 36 (17):6662–6667, 2003.

[8] E. H. A. de Hoog and R. H. Tromp. On the phase separation kinetics of an aqueous biopolymer mixture in the presence of gelation: the effect of the quench depth and the effect of the molar mass. *Colloid Surface A*, 213 (2-3):221–234, 2003.

[9] P. G. de Gennes. Reptation of a polymer chain in the presence of fixed obstacles. *J. Chem. Phys.*, 55 (2):572–579, 1971.

[10] J. M. Deutsch and T. L. Madden. The diffusion coefficient of a reptating polymer. *J. Chem. Phys.*, 91 (5):3252–3257, 1989.

[11] Enrico Carlon, Andrzej Drzewiński, and J. M. J. van Leeuwen. Crossover behavior for long reptating polymers. *Phys. Rev. E*, 64 (1):010801(R), 2001.

[12] M. Paeßens. *Reptation dynamics of short entangled polymers.* Diplom thesis, Rheinisch Westfälische Technische Hochschule, Aachen, 2001.

[13] R. E. Harrington. Opticohydrodynamic properties of high molecular weight DNA. III: The effects of NaCl concentrations. *Biopolymers*, 17:919, 1978.

[14] Oscar J. Lumpkin, Philippe Déjardin, and Bruno H. Zimm. Theory of gel electrophoresis of DNA. *Biopolymers*, 24:1573–1593, 1985.

[15] P. Déjardin, O. J. Lumpkin, and B. H. Zimm. *J. Polym. Sci., Polym. Symp.*, 73:67, 1985.

[16] Gary W. Slater and Jaan Noolandi. New biased-reptation model for charged polymers. *Phys. Rev. Lett.*, 55 (15):1579–1582, 1985.

[17] B. Widom, J. L. Viovy, and A. D. Defontaines. Repton model of gel electrophoresis and diffusion. *J. Phys. I France*, 1:1759–1784, 1991.

[18] T. A. J. Duke, A. N. Semenov, and J. L. Viovy. Mobility of a reptating polymer. *Phys. Rev. Lett.*, 69 (22):3260–3263, 1992.

[19] G. T. Barkema, C. Caron, and J. F. Marko. Scaling properties of gel electrophoresis of DNA. *Biopolymers*, 38:665–667, 1996.

[20] H. Hervet and C. P. Bean. Electrophoretic mobility of λ phage HIND III and HAE III DNA fragments in agarose gels: A detailed study. *Biopolymers*, 26:727–742, 1987.

[21] G. T. Barkema, J. F. Marko, and B. Widom. Electrophoresis of charged polymers: simulation and scaling in a lattice model of reptation. *Phys. Rev. E*, 49 (6):5303–5309, 1994.

[22] M. Olvera de la Cruz, J. M. Deutsch, and S. F. Edwards. Electrophoresis in strong fields. *Phys. Rev. A*, 33 (3):2047–2055, 1986.

[23] J. M. Deutsch and J. D. Reger. Simulation of highly stretched chains using long-range monte carlo. *J. Chem. Phys.*, 95 (3):2065–2071, 1991.

[24] Nicholas J. Rampino. Information concerning the mechanism of electrophoretic DNA separation provided by quantitative video-epifluorescence microscopy. *Biopolymers*, 31:1009–1016, 1991.

[25] Timothy D. Howard and G. Holzwarth. Fluctuations in the velocity of individual DNA molecules during agarose gel electrophoresis. *Biophys. J.*, 63:1487–1492, 1992.

[26] Thomas T. Perkins, Douglas E. Smith, and Steven Chu. Direct observation of tube-like motion of a single polymer chain. *Science*, 264:819–822, 1994.

[27] K. H. Meyer. *Z. Phys. Chem. (Leipzig)*, B44:383, 1939.

[28] Peter H. Verdier and W. H. Stockmayer. Monte Carlo calculations on the dynamics of polymers in dilute solution. *J. Chem. Phys.*, 36 (1):227–235, 1962.

[29] K. Kremer, A. Baumgärtner, and K. Binder. Collapse transition and crossover scaling for self-avoiding walks on the diamond lattice. *J. Phys. A: Math. Gen.*, 15:2879–2897, 1982.

[30] Kurt Kremer. Statics and dynamics of polymeric melts: A numerical analysis. *Macromolecules*, 16:1632–1638, 1983.

[31] Andrzej Kolinski, Jeffrey Skolnick, and Robert Yaris. Monte Carlo studies on the long time dynamic properties of dense cubic lattice multichain systems. I. The homopolymeric melt. *J. Chem. Phys.*, 86 (12):7164–7173, 1987.

[32] Andrzej Kolinski, Jeffrey Skolnick, and Robert Yaris. Monte Carlo studies on the long time dynamic properties of dense cubic lattice multichain systems. II. Probe polymer in a matrix of different degrees of polymerization. *J. Chem. Phys.*, 86 (12):7174–7180, 1987.

[33] Wen Bing Hu, Daan Frenkel, and Vincent B. F. Mathot. Lattice model study on the interplay of polymer phase transitions. *arXiv*, cond-mat/0206512:1–7, 2003.

[34] I. Carmesin and Kurt Kremer. The bond fluctuation method: A new effective algorithm for the dynamics of polymers in all spatial dimensions. *Macromolecules*, 21:2819–2823, 1988.

[35] I. Carmesin and Kurt Kremer. Static and dynamic properties of two-dimensional polymer melts. *J. Phys. France*, 51:915–932, 1990.

[36] H. P. Deutsch and K. Binder. Interdiffusion and self-diffusion in polymer mixtures: A Monte Carlo study. *J. Chem. Phys*, 94 (3):2294–2304, 1991.

[37] Wolfgang Paul, Kurt Binder, Dieter W. Heermann, and Kurt Kremer. Dynamics of polymer solutions and melts. Reptation predictions and scaling of relaxation times. *J. Chem. Phys.*, 95 (10):7726–7740, 1991.

[38] T. Kreer, J. Baschnagel, M. Müller, and K. Binder. Monte Carlo simulation of long chain polymer melts: Crossover from Rouse to reptation dynamics. *Macromolecules*, 34 (4):1105–1117, 2001.

[39] Pik-Jin Lai. Statics and dynamics of a polymer chain absorbed on a surface: Monte Carlo simulation using the bond-fluctuation model. *Phys. Rev. E*, 49 (6):5420–5430, 1994.

[40] M. Müller, J. P. Wittmer, and J.-L. Barrat. On two intrinsic length scales in polymer physics: topological constraints *vs.* entanglement length. *Europhys. Lett.*, 52 (4):406–412, 2000.

[41] K. E. Evans and S. F. Edwards. Computer simulations of the dynamics of highly entangled polymers. I. equilibrium dynamics. *J. Chem. Soc. Faraday Trans. 2*, 77:1891–1912, 1981.

[42] S. F. Edwards and K. E. Evans. Computer simulations of the dynamics of highly entangled polymers. II. static properties of the primitive chain. *J. Chem. Soc. Faraday Trans. 2*, 77:1913–1927, 1981.

[43] K. E. Evans and S. F. Edwards. Computer simulations of the dynamics of highly entangled polymers. III. Dynamics of the primitive chain. *J. Chem. Soc. Faraday Trans. 2*, 77:1929–1938, 1981.

[44] A. Baumgärtner, U. Ebert, and L. Schäfer. Segment motion in the reptation model of polymer dynamics. II. simulations. *J. Stat. Phys.*, 90 (5-6):1375–1400, 1998.

[45] M. Doi. *J. Polym. Sci. Polym. Phys. Ed.*, 21:667, 1983.

[46] G. T. Barkema and H. M. Krenzlin. Long-time dynamics of De Gennes' model for reptation. *J. Chem. Phys.*, 109 (15):6486–6489, 1998.

[47] Michael Rubinstein. Discretized model of entangled-polymer dynamics. *Phys. Rev. Lett.*, 59 (17):1946–1949, 1987.

[48] T. A. J. Duke. Tube model of field-inversion electrophoresis. *Phys. Rev. Lett.*, 62 (24):2877–2880, 1989.

[49] J. M. J. van Leeuwen and A. Kooiman. The drift velocity in the Rubinstein–Duke model for electrophoresis. *Physica A*, 184 (1-2):79–97, 1992.

[50] A. Kooiman and J. M. J. van Leeuwen. Reptation models for electrophoresis. *Physica A*, 194 (1-4):163–172, 1993.

[51] M. Prähofer and H. Spohn. Bounds on the diffusion constant for the Rubinstein–Duke model of electrophoresis. *Physica A*, 233:191–207, 1996.

[52] B. Akerman. Cyclic migration of DNA in gels: DNA stretching and electrophoretic mobility. *Electrophoresis*, 17 (6):1027–1036, 1996.

[53] Jean-Louis Viovy. Electrophoresis of DNA and other polyelectrolytes: Physical mechanisms. *Rev. Mod. Phys.*, 72 (3):813–872, 2000.

[54] Ryuzo Azuma and Hajime Takayama. Bond fluctuation method for a polymer undergoing gel electrophoresis. *Phys. Rev. E*, 59 (1):650–655, 1999.

[55] Justin Boileau and Gary W. Slater. An exactly solvable Ogsten model of gel electrophoresis: VI. Towards a theory for macromolecules. *Electrophoresis*, 22:673–683, 2001.

[56] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculation by fast computer machines. *J. Chem. Phys.*, 21 (6):1087–1092, 1953.

[57] J. M. Deutsch and T. L. Madden. Theoretical studies of DNA during gel electrophoresis. *J. Chem. Phys.*, 90 (4):2476–2485, 1989.

[58] T. A. J. Duke. Monte Carlo model of gel electrophoresis: steady state behavior. *J. Chem. Phys.*, 93 (12):9049–9054, 1990.

[59] T. A. J. Duke. Monte Carlo model of gel electrophoresis: response to field pulses. *J. Chem. Phys.*, 93 (12):9055–9061, 1990.

[60] Anatoly B. Kolomeisky. *One-dimensional nonequilibrium stochastic models, interface models, and their applications.* Ph.D. dissertation, Cornell University, 1998.

[61] Jaan Noolandi, Jean Rousseau, Gary W. Slater, Chantal Turmel, and Marc Lalande. Self-trapping and anomalous dispersion of DNA in electrophoresis. *Phys. Rev. Lett.*, 58 (23):2428–2431, 1987.

[62] Thomas L. Madden and Joshua M. Deutsch. Theoretical studies of DNA during orthogonal field alternating gel. *J. Chem. Phys.*, 94 (2):1584–1591, 1991.

[63] T. A. J. Duke and J. L. Viovy. Motion of megabase deoxyribonucleic acid during field-inversion gel electrophoresis: investigation by non-local monte carlo. *J. Chem. Phys.*, 96 (11):8552–8563, 1992.

[64] T. A. J. Duke and J. L. Viovy. Simulation of megabase DNA undergoing gel electrophoresis. *Phys. Rev. Lett.*, 68 (4):542–545, 1992.

[65] M. Widom and I. Al-Lehyani. Repton model of gel electrophoresis in the long chain limit. *Physica A*, 244 (1-4):510–521, 1997.

[66] M. E. J. Newman and G. T. Barkema. *Monte Carlo methods in statistical physics*. Clarendon press, Oxford, 1999.

[67] P. C. Hohenberg and B. I. Halperin. Theory of dynamic critical phenomena. *Rev. Mod. Phys.*, 49 (3):435–479, 1977.

[68] I. M. Lifshitz and V. V. Slyozov. The kinetics of precipitation from supersaturated solid solutions. *J. Phys. Chem. Solids*, 19:35–50, 1961.

[69] T. K. Kwei, T. Nishi, and R. F. Roberts. A study of compatible polymer mixtures. *Macromolecules*, 7 (5):669–674, 1974.

[70] F. S. Bates, S. B. Dierker, and G. D. Wignal. Phase behavior of amorphous binary mixtures of perdeuterated and normal 1,4-polybutadienes. *Macromolecules*, 19:1938–1945, 1986.

[71] Thein Kyu and Jeanne M. Saldanha. Phase separation by spinodal decomposition in polycarbonate/poly(methyl methacrylate) blends. *Macromolecules*, 21:1021–1026, 1988.

[72] Fu-Jya Tsai and John M. Torkelson. Phase separation of oligomeric polystyrene-polybutadiene blends as studied by excimer fluorescence. *Macromolecules*, 21:1026–1033, 1988.

[73] F. S. Bates, L. J. Fettes, and G. D. Wignall. Thermodynamics of isotopic polymer mixtures: poly(vinylethylene) and poly(ethylethylene). *Macromolecules*, 21:1086–1094, 1988.

[74] Frank S. Bates and Pierre Wiltzius. Spinodal decomposition of a symmetric critical mixture of deuterated and protonated polymer. *J. Chem. Phys.*, 91 (5):3258–3274, 1989.

[75] M. E. Fisher. Renormalization of critical exponents by hidden variables. *Phys. Rev.*, 176:257, 1968.

[76] D. J. Bergman and B. I. Halperin. Critical behavior of an Ising model on a cubic compressible lattice. *Phys. Rev. B*, 13 (5):2145–2175, 1976.

[77] A. L. Talapov and H. W. J. Blöte. The magnetization of the 3d Ising model. *J. Phys. A*, 29:5727–5733, 1996.

[78] J. M. D. Hill, B. McColl, D. C. Stefanescu, M. W. Goudreau, K. Lang, S. B. Rao, T. Suel, T. Tsantilas, and R. H. Bisseling. BSPlib: The BSP programming library. *Parallel Comput.*, 24:1947–1980, 1998.

[79] L. G. Valiant. A bridging model for parallel computation. *Commun. ACM*, 33 (8):103–111, 1990.

[80] R. H. Bisseling and W. F. McColl. Scientific computing on bulk synchronous parallel architectures. In *Proc. IFIP 13th World Computer Congress, Vol. I, North-Holland, Vol. 1*, pages 509–514, Amsterdam, 1994. North-Holland.

Samenvatting

(summary in Dutch)

Alle stoffen die we in de wereld om ons heen zien, zijn opgebouwd uit atomen, die onderling aan elkaar binden. Elk type atoom heeft de eigenschap dat het een sterke voorkeur heeft voor een bepaald aantal bindingen met andere deeltjes. Voor een waterstofatoom (H) is dat één binding, voor een zuurstofatoom (O) twee bindingen, voor een stikstofatoom (N) drie en voor een koolstofatoom (C) vier. De eenvoudigste molekulen die zo opgebouwd kunnen worden zijn waterstofgas (H_2), water (H_2O), amoniak (NH_3) en aardgas (CH_4). Atomen kunnen ook dubbele bindingen maken, bijvoorbeeld in zuurstofgas (O_2), koolstofdioxide (CO_2) en etheen/ethyleen (C_2H_4), of zelfs drievoudige bindingen, zoals in stikstofgas (N_2), ethyn (C_2H_2) en waterstofcyanide (HCN). Er zijn uitzonderingen op de regel, zoals bijvoorbeeld koolstofmonoxide (CO) en lachgas (N_2O), waarvan de structuur niet volgens deze regel verklaard kan worden.

Op ongeveer gelijke manier kunnen sommige soorten molekulen chemische bindingen vormen met andere molekulen. Als een molekuul twee andere molekulen kan binden, wordt het een monomeer genoemd. Twee monomeren kunnen dan binden tot een dimeer. Ook dit molekuul zal dan op twee plaatsen kunnen reageren met een ander monomeer of dimeer. Zo ontstaan ketens van drie, vier en meer herhaalde stukjes molekuul. Als zo'n molekuul lang wordt, noemen we dat een polymeer. Een polymeer is dus een langgerekt molekuul. Je kunt je zo'n polymeer voorstellen als een lange sliert spaghetti. Een mengsel of oplossing van polymeren kan een vloeistof-achtige toestand vormen. Dit wordt wel een smelt genoemd.

Behalve dat monomeren een langgerekt polymeer kunnen vormen, kunnen ook polymeren met zijtakken of netwerken van monomeren ontstaan. Zo'n netwerk kan dan een ruimtelijke structuur vormen. Je kunt je zo'n structuur voorstellen als een brij van spaghetti, waarbij spaghettisliertenhier en daar aan elkaar geplakt zitten. Een bekend voorbeeld is gelatine. Als gelatine wordt opgelost in warm water en vervolgens wordt afgekoeld, ontstaat er een stijve massa. De gelatinepolymeren veranderen bij een bepaalde temperatuur namelijk van vorm, en winden zich daarbij om elkaar heen. De polymeren die eerst allemaal onafhankelijk van elkaar konden bewegen vormen daarbij een stevige structuur. Deze massa gedraagt zich niet als een vloeistof, maar ook niet als een echte vaste stof. We noemen zo'n structuur een gel.

Dit proefschrift behandelt de manier van bewegen van een polymeer in een gel of in een smelt. In hoofdstuk 1 wordt uitgelegd wat de bewegingsvrijheid van een polymeer is. Het is bekend dat de beweging van een klein deeltje goed beschreven wordt door zogenaamde thermische fluctuaties. Het deeltje lijkt daardoor zo af en toe een zetje in een willekeurige richting te krijgen, wat ook wel Brownse beweging genoemd wordt. Als we de beweging van polymeren op deze manier willen beschrijven, moeten we het polymeer zien als een groot aantal aan elkaar gekoppelde polymeersegmentjes. We nemen dan aan dat de beweging van elk polymeersegmentje goed wordt beschreven door Brownse bewegingen, met de restrictie dat de (gemiddelde) afstand tussen de koppelingen van de polymeersegmentjes gelijk moet blijven.

Het gedrag van het hierboven beschreven abstracte model is vrijwel onafhankelijk van details van de beschrijving. Om simulaties te kunnen doen, moeten we echter een preciezer model kiezen. Dit kan op vele manieren worden gedaan; hoofdstuk 2 belicht een aantal populaire varianten van die modellen. Zoals we al gezegd hebben, kunnen we verwachten dat de preciese keuze van het model geen grote invloed heeft op de resultaten. Dit geeft de mogelijkheid om een zo eenvoudig mogelijk model te kiezen. In het bijzonder kunnen we de ruimte als een rooster beschrijven, waarbij de polymeersegmentjes van één roosterpunt naar een naburig roosterpunt kunnen bewegen, maar er niet tussenin kunnen zitten. Met dit soort eenvoudige modellen kan de diffusiecoëfficiënt uitgerekend worden. Deze beschrijft hoe snel een deeltje, in dit geval het hele polymeer, zich verplaatst. Omdat het deeltje kleine zetjes in willekeurige richtingen krijgt, kijken we naar het "gemiddelde gedrag", door de simulatie vele keren te herhalen. Als we de polymeren lang genoeg simuleren dan groeit de gemiddelde kwadratische verplaatsing evenredig met de tijdsduur van de simulatie. De evenwichtsconstante noemen we de diffusiecoëfficiënt. Dit is onder andere onderzocht voor polymeren die zich vrij kunnen bewegen, en voor polymeren die "opgesloten" zitten in een gel. In beide gevallen wordt de diffusiecoëfficiënt kleiner naar mate het polymeer langer is, maar in een gel neemt de diffusiecoëfficiënt sneller af dan wanneer het polymeer volledig vrij kan bewegen.

Hoofdstuk 3 beschrijft hoe we de belangrijkste roosterpolymeermodellen kunnen uitbreiden voor het beschrijven van gel-electrophorese. Dit is een experimentele techniek om polymeren van verschillende lengtes van elkaar te scheiden, door een kracht uit te oefenen op de polymeren; in dit geval door een electrisch veld aan te leggen. Een bekende toepassing is het maken van een zogenaamd DNA-profiel. DNA is een polymeer dat in elke cel van elk levende wezen te vinden is, en dat genetische informatie bevat. Deze code is opgeslagen in de volgorde van vier verschillende soorten monomeren, die in schijnbaar willekeurige volgorde geplaatst zijn en zo het polymeer vormen. Iedere menselijke cel heeft 46 verschillende DNA polymeren, de chromosomen, die gezamelijk alle erfelijke eigenschappen beschrijven. Het grootste deel van dit erfelijk materiaal is bij iedereen gelijk, maar kleine stukjes zijn uniek. Om een profiel te maken, wordt het DNA bewerkt met behulp van enzymen, die het DNA op bepaalde plaatsen in stukken knipt, en vervolgens op lengte “gesorteerd” met behulp van gel-electrophorese.

Op kleine schaal kunnen we het effect van het electrisch veld beschrijven door aan te nemen dat de deeltjes niet helemaal in een willekeurige richting worden geduwd, maar dat het net iets vaker voorkomt dat het deeltje met het veld mee beweegt, dan er tegen in. Het effect is dat het deeltje zich nu, gemiddeld gezien, in de voorkeursrichting verplaatst. De snelheid waarmee het deeltje beweegt afhangt van de aangelegde kracht, en van de diffusiecoëfficiënt. In hoofdstuk 4 beschrijf ik hoe ik de snelheid van polymeren in een gel onder invloed van een elektrisch veld heb uitgerekend. Het model wat we hebben gebruikt voor de beschrijving van de polymeren is het “cage-model”. Zoals verwacht is de snelheid van de polymeren evenredig met het aangelegde veld als dat niet al te groot is. Als het veld sterker wordt gemaakt, oriënteert het polymeer zich in de richting van het veld. Het gevolg daarvan is dat het polymeer zich gedraagt als een keten deeltjes die onafhankelijk van elkaar bewegen. Het blijkt dat de snelheid dan kwadratisch afhangt van de sterkte van het aangelegde veld, en onafhankelijk is van de lengte van het polymeer. Als het veld nog sterker wordt gemaakt verplaatst de “achterkant” van het polymeer zich sneller in de richting van het aangelegde veld dan de “voorkant”. Hierdoor ontstaan opeenhopingen van een teveel aan polymeer, die vervolgens kunnen uitstulpen. Deze uitstulpingen noemen we “hernias”. Net als de rest van het polymeer richten deze hernias zich in de richting van het elektrische veld en zullen dan niet gemakkelijk verdwijnen. De polymeersegmentjes in de hernia bewegen namelijk liever met het veld mee, terwijl de hernia alleen kan oplossen als de segmentjes tegen het veld in bewegen. Als de achterkant van het polymeer een hernia inhaalt ontstaan polymeren met een U-vorm, waarbij beide einden in de richting van het veld wijzen, maar het polymeer in de gel vast blijft zitten totdat alle segmentjes in één van de uiteinden naar de andere kant zijn getransporteerd. Dit is een uiterst moeizaam proces, en de snelheid van het polymeer is al die tijd verwaarloosbaar.

In hoofdstuk 5 wordt, net als in hoofdstuk 4, de snelheid van cage-polymeren onderzocht. Hier echter met een methode die exact is. Dit beperkt de maximale lengte waarvoor de snelheid kan worden uitgerekend, maar heeft het voordeel dat de numerieke fout in de gevonden snelheden zeer klein is, zodat er een betere analyse van de resultaten kan worden gemaakt. Een polymeer in het cage-model wordt beschreven door één van de eindpunten te nemen en vervolgens de richting van de opeenvolgende segmentjes van het polymeer te geven. We hebben voor polymeren tot en met lengte 15 de waarschijnlijkheid van alle mogelijke configuraties berekend (voor lengte 15 zijn dat er 6^{14}, ofwel 78 364 164 096). Dit was mogelijk doordat we een relatief eenvoudige beschrijving van alle symmetrieën konden vinden (zodat er nog maar 5 154 859 unieke configuraties overbleven).

Tot en met hoofdstuk 5 hebben we alleen gesproken over een enkel polymeer in een gel. In hoofdstukken 6 en 7 willen we echter een systeem beschrijven waarin twee soorten polymeren zitten, die in een gesmolten toestand (smelt) verkeren. Dat wil zeggen dat de polymeren niet opgesloten zitten in een rooster, zoals we de gel hebben beschreven, maar vrij kunnen bewegen, en alleen gehinderd worden door andere polymeren. In een goede beschrijving mogen verschillende segmenten van polymeren niet op dezelfde plaats zitten. Ook moeten naburige segmenten van verschillend type zich anders gedragen dan naburige segmenten van gelijk type. Wij hebben gekozen voor een afstotende interactie tussen segmenten van verschillend type, zodat de polymeren domeinen vormen van gelijke polymeren.

In hoofdstuk 6 beschrijven we een model waarmee we deze fasescheiding efficiënt konden simuleren. Deze simulaties richten zich op metingen van de domeingrootte. De theoretische verwachting is dat de gemiddelde domeingrootte groeit als de derde-machts wortel van de verstreken tijd. Dit blijkt inderdaad in overeenstemming te zijn met de resultaten van de simulaties. Voorzover wij weten zijn dit de eerste simulaties die dit gedrag voor afstotende polymeermengsels aantonen.

Tenslotte, in hoofdstuk 7, gebruiken we hetzelfde polymeermodel om te onderzoeken hoe de evenwichtstoestand van het mengsel eruit ziet als de fasescheiding voltooid is. In hoofdstuk 6 waren we geïnteresseerd in het tijdsafhankelijk proces van domeingroei en moesten ons beperken tot realistische stappen, zoals het verplaaten van één segmentje over een afstand van één roosterafstand. Om evenwichtseigenschappen te bepalen is dit niet nodig. Het blijkt voor de efficiëntie van de simulatie zeer veel te helpen om ook het veranderen van soort van een polymeerketen toe te voegen aan de mogelijke stappen.

De polymeren in een polymeeroplossing zijn meestal niet allemaal even lang. Er zijn langere en kortere polymeren in het mengsel. Als we een ontmengde oplossing met twee polymeertypen bekijken zullen we een fase vinden met vooral polymeren van het het ene type, en een andere fase met vooral polymeren van het andere type. Het blijkt echter dat er altijd wel polymeren van het vreemde type achterblijven, ook als we het systeem heel lang tot rust laten komen. Uit experimenten is gebleken dat vooral korte polymeren in de vreemde fase te vinden zijn. Fysisch gezien is dit eenvoudig te verklaren als je bekend bent met de begrippen (meng-)entropie en energie, omdat de entropie-winst van een deeltje in de vreemde fase onafhankelijk is van de lengte, terwijl de verhoging van de totale energie evenredig is met de lengte van het polymeer. We bewijzen in dit hoofdstuk dat de zogenaamde Flory-Huggins theorie, toegepast op het roostermodel, de experimenten goed beschrijft. Bovendien hebben we de Flory-Huggins theorie aangepast zodat het ook kleine systemen kan beschrijven, en de resultaten vergeleken met de simulatieresultaten. Ook dit blijkt goed te werken.

Dankwoord

(acknowledgements)

In dit dankwoord wil ik graag iedereen bedanken die op één of andere manier heeft bijgedragen aan dit proefschrift. Op het wetenschappelijke vlak bedank ik in de eerste plaats Gerard Barkema voor zijn dagelijkse begeleiding bij het tot stand komen van de publicaties die de basis vormen voor dit proefschrift. Ook bedank ik Rob Bisseling, in het bijzonder voor het leveren van het bewijs van de correctheid van de kink-representatie van het cage model. Ook bedank ik Henk van Beijeren voor zijn kritische blik en vele opbouwende op- en aanmerkingen over de tekst van dit proefschrift.

Verder hebben vele mensen indirect bijgedragen aan dit proefschrift door voor een prettige werksfeer op het instituut te zorgen. Zowel in de periode dat het Instituut voor Theoretische Fysica nog in het BBL gevestigd was, als na de verhuizing naar het Minnaertgebouw. In het bijzonder wil ik daarbij mijn vaste kamergenoot Rembert Duine noemen. Ook voor de secretaresses Leonie, Natasja, Biene, Geertje en Wilma, de mede-AIO's en OIO's en overige medewerkers: bedankt!

I also like to thank Anatoly Kolomeisky for inviting me to visit him at Rice University. Tenslotte wil ik ook Els de Hoog, Marijke Edelman, Hans Tromp en Erik van der Linden bedanken voor de prettige samenwerking in het kader van Softlink.

Curriculum Vitae

Ik ben geboren op 9 mei 1975 in Rotterdam en opgegroeid in Giessenburg. In 1994 behaalde ik mijn VWO diploma op het Gymnasium Camphusianum in Gorinchem. Aansluitend begon ik aan de Universiteit Utrecht mijn natuurkundestudie, met als hoofdrichting het indertijd nieuwe programma computationele natuurkunde. In de laatste jaren van mijn studie heb ik, als student-assistent, de computerpractica van de natuurkundefaculteit begeleid. In de cursussen Inleiding computergebruik, Mathematica en Programmeren in C werden eenvoudige natuurkundige problemen doorgerekend en in de cursus LabVIEW werden eenvoudige experimentele opstellingen aangestuurd en de meetresultaten geanalyseerd. Als eindonderzoek werkte ik onder leiding van Dr. G. T. Barkema en Dr. R. H. Bisseling aan numeriek exacte berekeningen van de driftsnelheid van roosterpolymeren onder invloed van een aangelegde kracht. In oktober 1999 rondde ik deze scriptie, en daarmee mijn studie, af. Daaropvolgend ben ik, weer onder leiding van Dr. G. T. Barkema aan de Universiteit Utrecht, als onderzoeker in opleiding bij de stichting F.O.M. in dienst getreden. Resultaten van het onderzoek dat ik in deze periode heb gedaan zijn gepubliceerd in vijf artikelen. Deze artikelen vormen de basis voor dit proefschrift. Gedurende de eerste jaren van het promotiewerk heb ik weer computercursussen gegeven, maar ook practica voor de studies fundamentele biomedische wetenschappen en medische biologie, waarin onder andere microscopische onderzoekstechnieken werden behandeld. Op 1 januari 2004 begin ik aan mijn nieuwe baan aan de Universität des Saarlandes, bij Prof. dr. H. Rieger in de groep Theoretische Fysica, in Saarbrücken, Duitsland.