Monte Carlo simulation of 2D polymer thermodynamics using the pruned-enriched Rosenbluth method

Second project for Computational Physics

D.S. van Holthe tot Echten, J.J. Slim, and S.H.O. Zwetsloot

Faculty of Applied Sciences, Delft University of Technology, The Netherlands

April 9th, 2016

ABSTRACT

The goal of this simulation was to simulate the thermodynamic equilibrium properties of 2D polymers under varying conditions. This simulation was done using a pruned-enriched Rosenbluth method for Lennard-Jones polymers growing up to a length of 250 beads. The thermodynamic behaviour of the system was captured by using the Boltzmann weights for adding new polymer segments. During these simulation the temperature of the system and the linear force on the polymer was varied. The temperature varied simulations show that the Lennard-Jones attractive potential on the polymer is dominant for low temperatures. For higher temperature the repulsive part of the repulsive part of the Lennard-Jones potential becomes dominant. Increasing the force on the polymer shows the end to end distance of the polymer exponentially approaching an asymptote at the contour length of the polymer. Varying both the temperature and the force show that increasing force becomes less effective for higher temperatures.

Key words. polymers – Monte Carlo simulation – PERM – Pruned-enriched Rosenbluth

1. Introduction

Although the laws of physics allow us to describe the dynamics of most (classical) systems with arbitrary accuracy, the numerical estimation of the behaviour of a many-particle system quickly becomes infeasible as the number of degrees of freedom grows. For systems with only short-range interactions this may be overcome by simulating only a very small part of a given system in such a way that the results can be extrapolated to larger systems. This is for example done in molecular dynamics simulation, where periodic boundary conditions are used to mimic an infinite system, together with a minimal image convention to restrict the necessary computational resources.

Another solution to this problem may be found by abandoning the wish of simulating the dynamical behaviour of the system's degrees of freedom directly, and substituting these dynamics by artificial dynamics that are easier to simulate. This means that we can not estimate *dynamical* quantities from the simulation anymore but if the substitution of dynamics is done carefully, the result of statistical estimations of *static* quantities can be kept the same.

In general, static quantities may be calculated from an integral over the phase space of the system:

$$A = \langle A(X) \rangle = \int \rho(X) A(X) \, \mathrm{d}X \tag{1}$$

where A is the quantity of interest which depends on the system microstate X. Each microstate is weighed in the integral with the probability $\rho(X)$ of the occurence of microstate X. To be able to calculate static quantities from a system driven by artificial dynamics, we must make sure that the probability $\rho_A(X)$ of each microstate occurring in the 'artificial' simulation coincides with the probability $\rho(X)$ of that microstate occurring in the real system.

1.1. Monte Carlo integration

This idea of substituting the real dynamics of a system by artificial dynamics based on random numbers is central to the so-called Monte Carlo methods. In a typical MC simulation, new system microstates are continuously generated according some distribution $\rho'(X)$. The dynamical evolution of the microstates is completely ignored in principle. If we make sure that this distribution $\rho'(X)$ equals the real distribution $\rho(X)$, this method is called *importance sampling* MC. In this case, the integral (1) can then be estimated by calculating the expectation value of the quantity of interest A over the generated microstates:

$$\bar{A} = \sum_i A(X_i) \tag{2}$$

The most naive implementation of this scheme would be to generate microstates at random and accepting them with probability $\rho(X)$. However, the probability $\rho(X)$ may be very non-uniform over state space, leading to the rejection of the majority of the generated microstates. Such a scheme is therefore very computationally wasteful.

1.2. Markov Chain models

In an attempt to generate system microstates more efficiently, a Markov chain model can be employed. Note that the Rosenbluth method we will finally use to simulate polymer thermodynamics is not an ergodic Markov chain model per se. It does however share the idea that generating new 'acceptable' samples from previous 'acceptable' samples is more efficient than generating new 'acceptable' samples from scratch. This section is therefore included as a general reference.

A Markov chain model consists of a given state space and a set of transition probabilities between the states. These can be expressed by the transition matrix P of probabilities $P_{ij} = P(i \to j)$ of the transition from state i to state j.

Such a model is initialized in a certain state X_0 and allowed to evolve step-wise. In every step a transition occurs that is selected from the set of possible transitions with probabilities given by the transition matrix P. The resulting progression $\{X_t\}$ through the model's state space is called a Markov chain.

The probability to find the model in a certain state X after step t is given by the function $\rho(X, t)$. For a discrete state space with $N = \#X$ states we can express this as an N-dimensional vector $\boldsymbol{\rho}(t)$. The evolution of this vector is governed by the following equation:

$$\boldsymbol{\rho}(t+1) = P\boldsymbol{\rho}(t) \tag{3}$$

If the probability vector $\boldsymbol{\rho}_s(t)$ is an eigenvector of the transition matrix P, we see that the probability distribution over the various states X remains the same in the subsequent time steps:

$$\boldsymbol{\rho}_s(t+1) = P\boldsymbol{\rho}_s(t) = \lambda\boldsymbol{\rho}_s(t) = \boldsymbol{\rho}_s(t) \tag{4}$$

Note that the eigenvalue λ of an eigenvector of P necessarily has to be one because the transition matrix preserves probability. Such a probability distribution vector $\boldsymbol{\rho}_s$ is called the stationary or invariant distribution of the Markov chain model.

If the probability distribution of a Markov chain converges towards the stationary distribution $\boldsymbol{\rho}_s$ as the number of evolution steps the system undergoes tends to infinity, the model is called *ergodic*. The Markov chain has to satisfy two conditions to be ergodic: 1) it has to be *connected*; that is, every state that may occur in the model must be accessible from every other possible state in a finite number of steps; and 2) it has to be *aperiodic*, meaning that none of the possible states have a periodicity in the number of steps in which the system can return to that state.

1.3. Standard Markov Chain Monte Carlo methods

The trick of using Markov chain models for Monte Carlo integration lies in choosing the transition matrix P such that the stationary distribution $\boldsymbol{\rho}_s$ equals the microstate distribution $\rho(X)$ of the system we want to simulate. We may then use the microstates generated by the Markov chain to calculate the phase space integral (1) with the sum (2). In this scheme, every state we generate (after the initial equilibration of the Markov chain towards the stationary distribution) contributes to the calculation the expectation value, making this method much less computationally wasteful than the naive Monte Carlo integration described earlier.

2. Polymer dynamics

In this research, the thermodynamic equilibrium properties of dilute 2D polymer chains are investigated. A polymer molecule, for example DNA or various plastic molecules, consists of a long chain of monomer molecules that are bonded together chemically. In general, these chemical bonds are not extremely rigid and neighbouring parts of the polymer may be rotated with respect to each other.

If these polymers are suspended in a liquid they can move around more or less freely and thermally induced random rotations between the monomers will cause the polymer to curl up. Different parts of the chain may then interact with each other as they are brought closer together by this coiling effect.

This interaction may be attractive or repulsive, depending on the specific conditions and separation distance. Typically, a Van der Waals interaction is assumed between the polymer parts, modelled by a Lennard-Jones pair potential. Additionally, an interaction between polymer parts may be mediated by solvent molecules. If the solvent is good for the particular polymer under investigation, the free energy of a polymer part that is surrounded by solvent molecules is lower than the free energy of a polymer part that is surrounded by other polymer parts. This translates into an effective repulsion between polymer parts.

2.1. The bead model

In order to arrive at a model that can be used to simulate such dilute 2D polymers, we assume that the polymer consists of segments (which may not perfectly coincide with monomers) that are bound together by chemical bonds that can be rotated freely with no energy penalty for the rotation per se. These segments are called 'beads' and are modelled as point particles. The distance between adjacent beads is assumed to be fixed by the chemical bond to a distance d. Furthermore, we assume that the beads interact pairwise with each other according to a Lennard-Jones potential, which describes the combined effect of both the Van der Waals interaction and the solvent-mediated interaction. The interaction assigns an energy to every polymer configuration.

In this model, a polymer X is completely described by the list of positions $\boldsymbol{r}_i$ of all N beads. The effect of the chemical bonds can be described by the following constraint:

$$|\boldsymbol{r}_{i+1} - \boldsymbol{r}_i| = d \tag{5}$$

For simplicity, the units in this research have been chosen such that $d = 1.0$ in reduced units.

The energy of a polymer is given by:

$$E(X) = \sum_{\substack{i,j \\ i<j}}^{N} V_{LJ}(|\boldsymbol{r}_i - \boldsymbol{r}_j|) \tag{6}$$

where the $i < j$-constraint in the sum prevents the double counting of pairs and V_{LJ} is the Lennard-Jones pair potential we use, given by:

$$V_{LJ}(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \tag{7}$$

The parameters σ and ϵ describe the typical interaction length and energy scales respectively and may be determined experimentally. In this research $\sigma = 0.8$ and $\epsilon = 0.25$ have been used (in reduced units), as suggested by (1).

2.2. Model statistics

The probability of the occurrence of a certain polymer configuration X in an (N,T) or canonical ensemble is proportional to the Boltzmann factor $f(X)$ of that configuration, given by:

$$\rho(X) \propto f(X) = e^{-E(X)/k_B T} \tag{8}$$

Throughout this research we use reduced units with $k_B = 1$.

Note that the energy of polymer X can be written as the sum of the 'addition energies' of adding the beads one-by-one:

$$E(X) = \sum_{i=1}^{N} E_{add}(\boldsymbol{r}_i; \boldsymbol{r}_{i-1}, \ldots, \boldsymbol{r}_1) \tag{9}$$

where E_{add} indicates the energy needed to add a bead at $\boldsymbol{r}_i$ to a chain of length $i-1$ that has already been laid out as $\boldsymbol{r}_{i-1}, \ldots, \boldsymbol{r}_1$:

$$E_{add}(\boldsymbol{r}_i; \boldsymbol{r}_{i-1}, \ldots, \boldsymbol{r}_1) = \sum_{j=1}^{i-1} V_{LJ}(|\boldsymbol{r}_i - \boldsymbol{r}_j|) \tag{10}$$

This allows us to write the following for (8):

$$\rho(X) \propto e^{-E(X)/k_BT} = e^{-\sum_{i=1}^{N} E_{add}(\boldsymbol{r}_i)/k_BT} = \prod_{i=1}^{N} e^{-E_{add}(\boldsymbol{r}_i)/k_BT} \tag{11}$$

This enables us to calculate static equilibrium properties of the modelled polymers by using equation (1). If we construct an algorithm that generates polymers distributed according to $\rho(X)$ we can then use equation (2).

2.3. Relevant static properties

Interesting static properties of the polymer include the end-to-end distance of the polymer and the gyration radius.

2.3.1. End-to-end distance

The end-to-end distance R of an N-bead polymer is simply the distance between the last and the first bead:

$$R = |\boldsymbol{r}_N - \boldsymbol{r}_1| \tag{12}$$

This quantity is for example interesting in biological applications, where the end-points may be connected to various cell components. For random walks, the following general law relates the length to the number of links N:

$$R \propto N^{\nu} \tag{13}$$

For a non-selfavoiding random walk, the exponent ν has value $\nu = 0.5$ while in the self-avoiding case the exponent value is $\nu = 0.75$

2.3.2. Gyration radius

The gyration radius of an N-bead polymer is a measure for the distribution of the polymer in space. Polymers with a low gyration radius are more coiled up than polymers with a higher gyration radius. The definition of the gyration radius R_g is given by:

$$R_g^2 = \frac{1}{N} \sum_{l=1}^{N} (\boldsymbol{r}_l - \boldsymbol{r}_{mean})^2 \tag{14}$$

Here r_l is the position of bead l and r_{mean} is the mean position of all the beads of the polymer.

3. Pruned-enriched Rosenbluth method (PERM)

The *Rosenbluth method* is originally an algorithm that generates realizations of self-avoiding random walks, by maintaining a population of random walks (the samples) and adding a new step to every sample at every iteration. Because of the self-avoidance criteria, the number of possibilities to add a new bead may vary. This influences the statistics of the population, and needs to be compensated by weighting every sample with a factor that indicates how many possible walks were blocked because of self-avoidance criteria (2). The Rosenbluth method can be easily adapted to accommodate polymer dynamics using equation (11)

A common problem with Rosenbluth simulations for longer chains is that a lot of computing power is spent on samples with relatively low weight that don't contribute much to the final statistics. A solution for this problem lies in the *pruning* of low-weight configurations and the subsequent *enriching* (duplication) of high-weight configurations to take their place. This is called the *pruned-enriched Rosenbluth method*.

3.1. General overview of the algorithm

The algorithm used to study polymer thermodynamics and obtain the results presented in this research has the following general layout. Each of these items will be discussed briefly in the following sections.

- Initialisation of the population
- A general loop that runs until the maximum length is reached:
  - Addition of a single bead across the population
  - Measurements at the current polymer length
  - Pruning of low-weight polymers
  - Enrichment of high-weight polymers
- Storage of the results

3.2. Population initialisation

Initially the entire population (of typically 10.000 - 50.000 polymers) is initialised with a single bead at position $(0, 0)$. If the problem would be rotationally invariant (which it is if there are no external forces in play) we may proceed immediately to place a second bead at $(d, 0)$. As we do however experiment with external forces that break rotational symmetry, this second bead is not immediately added.

3.3. Addition of single bead across the population

Subsequently, in every iteration a single bead is added to all the polymers in the current population. The only degree of freedom we have for each polymer is the angle at which the next bead will be placed with respect to the previous one, as the distance between the beads is fixed at d. Ideally, new angles are drawn from a Boltzmann distribution to capture the thermodynamics of the system accurately.

To do so, for every possible placement angle θ the addition energy E_{add} has to be calculated. This is computationally prohibitive, and as a solution we discretize the possible placement angles θ, in this case into $N_\theta = 6$ evenly spaced placement angles that are offset from their 'natural' positions by a random angle drawn uniformly from $[0, 2\pi/N_\theta]$

Using this we calculate the position of our N_θ candidate bead positions for the next bead.

$$\boldsymbol{r}_{i+1,j} = \boldsymbol{r}_i + d \begin{bmatrix} \cos(\theta_j) \\ \sin(\theta_j) \end{bmatrix} \tag{15}$$

For each of these candidate bead positions the addition energy is determined and the Boltzmann weight $w_j^{(l)} = \exp(-E_{add}/k_BT)$ is calculated. With these weights one of the candidate bead positions is selected as the next bead using a *roulette-wheel algorithm* where one of the candidates is selected with probability:

$$P_j^{(l)} = \frac{w_j^{(l)}}{\sum_{j=1}^{N_\theta} w_j^{(l)}} = \frac{w_j^{(l)}}{W^{(l)}} \tag{16}$$

Thermodynamically, the probability of occurrence of a certain polymer X is given by the product of the Boltzmann weights w (cf. (11)):

$$f_{therm}(X) = \prod_{l=1}^{N} w_{\theta}^{(l)} \tag{17}$$

In our scheme however, this probability is given by the product of selected $P_j^{(l)}$'s:

$$f_{Rblth}(X) = \prod_{l=1}^{N} P_j^{(l)} = \prod_{l=1}^{N} \frac{w_j^{(l)}}{W^{(l)}} \tag{18}$$

We see that the probabilities are off by a factor $\prod_l W^{(l)}$. To compensate for this error, every polymer must be weighted with this *weight factor* when calculating statistical averages. This weight factor serves the same role as the Rosenbluth self-avoidance compensation factor in the original Rosenbluth algorithm. These polymer weight factors are updated and tracked throughout the algorithm

3.4. Measuring polymer quantities

After a new bead has been added to every polymer, interesting static quantities may be calculated. The polymer weight factors are taken into account when calculating these averages.

3.5. Pruning of low-weight polymers

To spend computing time more efficiently, polymers with low weights are pruned using criteria as suggested by (1). First, the average polymer weight AvWeight across the population is determined. From this average, a lower limit is determined for every polymer using its weight at step 3 (at which the first independent bead is added) with the following formula:

$$\text{LowerLimit} = \alpha \,\text{AvWeight} \,/\, \text{Weight3} \tag{19}$$

α was selected to be 1.2. From the k polymers with weights below this lower limit, $k/2$ polymers are selected randomly to be pruned (removed from the population) while the other $k/2$ polymers have their weight doubled to keep the overall polymer weight at the same level.

3.6. Enrichment of high-weight polymers

To compensate this decimation of the polymer population, successful polymers are copied to take the place of the pruned polymers. The procedure we follow here in this research is slightly different than in (1); we don't work with an upper limit above which polymers are duplicated. Instead we order the polymers by weight and select the top $k/2$ polymers. These polymers are duplicated and put in place of the $k/2$ pruned polymers. This ensures that the population size remains stable. The weight of both polymer duplicates is halved to make sure the population weight remains equal.

In most simulation this lead to a steady pruning rate of about 3% of the population at every iteration.

3.7. Growing until the maximum length

These three steps - addition, measurement and pruning/enrichment - are repeated until a specified maximum length is reached. Throughout this research, a maximum length of 250 beads has been used. The simulation of a population of 10.000 polymers took about 5 minutes, the simulation of 50.000 polymers took about 25 minutes.

4. Results

During our simulation we first determined the end-to-end distance distribution for polymers on increasing length under regular conditions with a temperature of $T = 1.0$. Then we started changing parameters to see their effect on the final distribution. The parameters that were changed were the temperature and a simulated force on the polymer in the horizontal direction.

4.1. Initial results

Initially we ran a simulation under regular conditions to verify our model. We made a simulation with 10.000 polymers. According to equation (16) literature shows that the end to end distance of the polymer related to the length of the polymer should be of the following form for high temperatures:

$$R(N) = c_0 \cdot N^{0.75} \tag{20}$$

In figure 1 our simulation results are shown plotted together with the fit. This figure shows a good correlation between our simulation results and literature values, since the value ν of the fit (0.742) corresponds to the theoretical value of 0.75. The deviation in the beginning is a kind of edge effect as the first link always has length $d = 1.0$. With this verification of our model, we start changing parameters to see their effect on the final distribution.

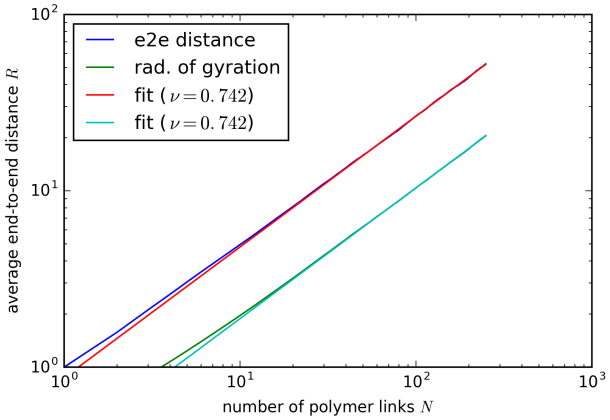

Fig. 1. Log-log plot of the end to end distance and the gyration radius versus the bead count of a family of 10.000 polymers at $T = 1.0$. The blue line shows our simulation results. The red line shows a fit of the form aN^{ν}. For self-avoiding walks ν is expected to be around 0.75, which is nicely matched in this simulation. The gyration radius follows a similar power law with respect to the number of beads. Error bars are not shown here to prevent visual clutter, but the relative statistical errors encountered are consistently on the order of 0.3%

4.2. Temperature variation

When we increase the temperature we expect the end-to-end distance also to increase, since the attractive part of the Lennard-Jones potential will be dominated by the thermal energy. In figure 2 it is shown that indeed the end-to-end distance and the

gyration radius increase for higher temperature. At $T = 1.0$ the lengths saturate: when the repulsive part of the potential already dominates, further increase of the temperature will not result in differences of the lengths. Figure 2 shows a clear distinction between a low-T and a high-T regime with a transition in between.

The change of the behaviour can also be seen in figure 3, figure 4 and figure 5 where the polymers at low temperature tend to curl up, while the polymers at higher temperature develop further from the origin.

Interesting to notice is that at very low temperatures, different polymers are very much correlated and the phase space can not be sampled efficiently anymore by the PERM algorithm. The attractive part of the potential results in dependency of the positions of the beads and copying (enrichment) of these polymers result in correlation between polymers. This causes the strong fluctuations in the low-T regime.

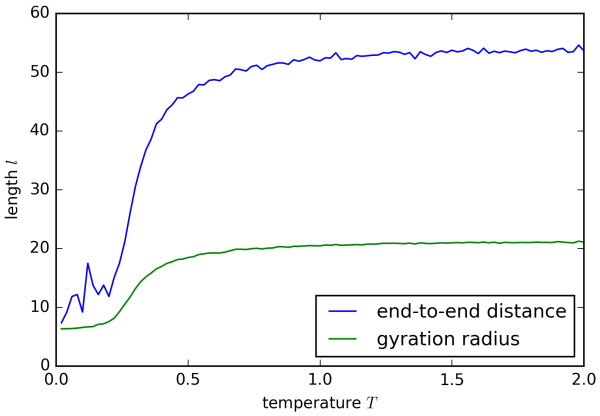

Fig. 2. Average end to end distance and gyration radius for collections of 50.000 polymers of 250 beads generated by the PERM algorithm at temperature ranging from 0.02 to 2.00. Two different regimes can be distinguished depending on the temperature. In the high-temperature regime the attractive potential between polymer segments is no longer relevant with respect to the thermal energy. The behaviour of the polymer is then dictated by the repulsive part of the Lennard-Jones potential, which is relevant at every temperature; this leads to minimal temperature dependence in the high T regime. In the low-temperature regime the attractive forces cause the polymer to curl up, resulting in a markedly lower end-to-end distance and gyration radius.

4.3. Force variation

Finally we applied an external force by adding a work term $W = -F\Delta x$ to the addition energy E_{add} of every bead, where Δx is the amount of distance covered by the next link in the force (x) direction. This is equivalent to pulling on the free end of the polymer with a force F in the x-direction while fixing the other end of the polymer at the origin

We varied the force with a value between $F = 0.0$ and $F = 8.0$. As expected this resulted in an increasingly growth of the polymer in one direction. Figure 6 shows a sample of polymers with a force in the positive x-direction applied to them. The plot shows that the polymers have a strong preference to growing into the same direction of the applied force.

The result of this simulation can be be seen in figure 7. As expected, the end to end distance of the polymer approaches log10(250) asymptotically. Under this assumption we tried to fit

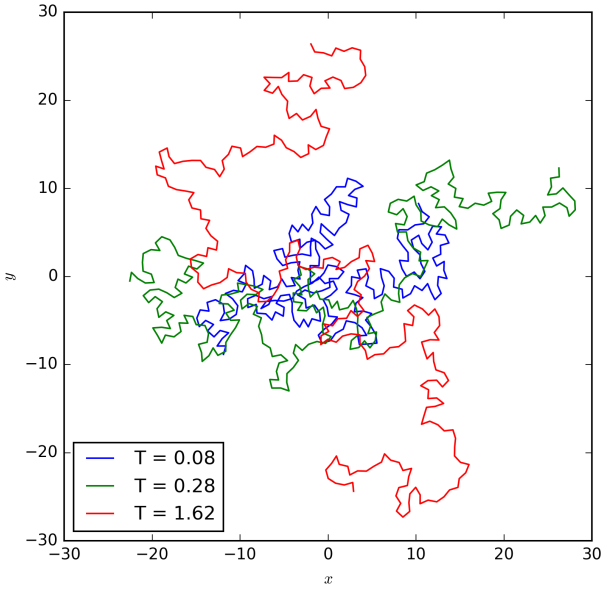

Fig. 3. Three arbitrarily picked 250-bead polymer realizations at increasing temperatures. For the $T = 0.08$ polymer a strong favour for coiling around itself can be seen, indicating the relevance of the attractive Lennard-Jones forces at this temperature. At higher temperatures, these attractive forces become smaller compared to the thermal excitation energy. For the $T = 0.28$ polymer this leads to a partial breakdown of the coiling state and an increased end-to-end distance. For the $T = 1.62$ the energy gained from the attractive potentials is fully negligible with respect to the thermal energy, breaking down the stability of the coiling state completely.

the simulation reuslts to a function of the form:

$$E2E(F) = 250 - (250 - 250^{0.75}) \cdot e^{pol(F)} \tag{21}$$

In this function the 250 is the asymptotic value which the function should approach. The $250 - 250^{0.75}$ term is to account for the literature distance of $250^{0.75}$ when the force on the polymer is zero. Pol(F) is a polynomial function of F which has to be determined. Using a least-square fit in the semi-log domain we calculated the best fit for the polynomial function of F. This resulted in the following function:

$$pol(F) = 0.273F^{0.1} + F^{0.5} \tag{22}$$

4.4. Comparison with 2D freely jointed chain model

We also made an attempt to compare the force-extension curves we obtained at different temperatures to FE-curves predicted by theory. Figure 8 shows two simulated FE-curves and the corresponding FE-curves for 2D freely-jointed chain models (which don't include any interactions and are therefore a priori a bad match for our simulations, but they are analytically solvable) (5). We see that the extra LJ repulsion dominates at high temperature, showing a tendency of the polymer to 'open up' at lower forces than its FJC counterpart. The LJ attraction at low temperatures keeps the polymer closer together at higher forces however.

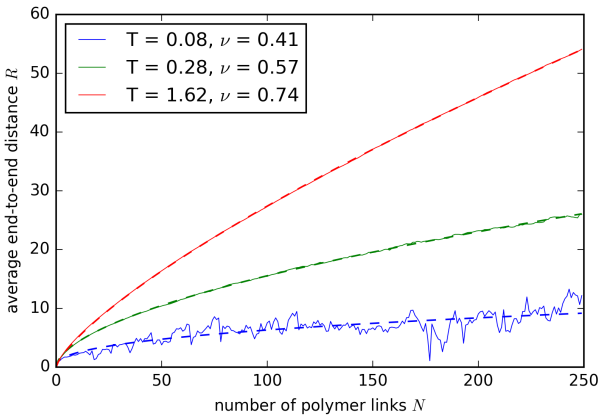

Fig. 4. Average end-to-end distances for three collections of 50.000 polymers as a function of the number of beads. Solid lines indicate simulation results, dashed lines indicate fit results to a power law with exponent ν. The three collections are generated at three different temperatures: $T = 0.08$ in the low-T regime, $T = 0.28$ in the transition regime and $T = 1.62$ in the high-T regime. We see that the length of the high-T polymers matches with the SAW $\nu = 0.75$-law, but this does not hold for the lower temperatures. At low temperatures (smaller than 0.10), different polymers are very much correlated and the phase space can not be sampled efficiently anymore by the PERM algorithm. This causes irregularities at low temperature.

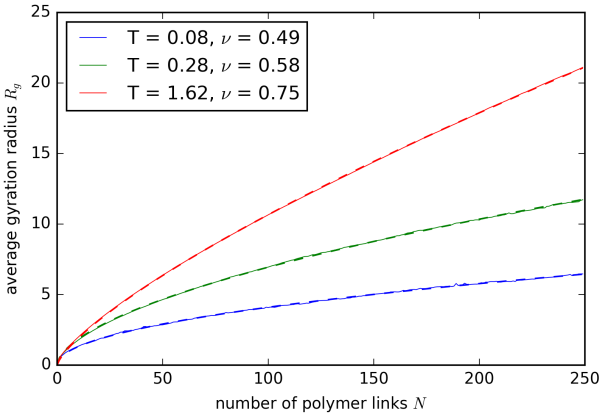

Fig. 5. Average gyration radii for three collections of 50.000 polymers as a function of the number of beads. Solid lines indicate simulation results, dashed lines indicate fit results to a power law with exponent ν. The three collections are generated at three different temperatures: $T = 0.08$ in the low-T regime, $T = 0.28$ in the transition regime and $T = 1.62$ in the high-T regime. It is very interesting to note that the fluctuations visible in the low temperature end-to-end distance graph of figure 4 are absent in the low temperature gyration radius.

4.5. Combining temperature and force variation

To see the effect of force on the polymers for different temperatures, we did a simulation with 10.000 polymers per simulation ranging over a temperature range of 0.4-4.0 and a force range of 0.0-3.0 both increasing with steps of 0.2. The results of this simulation are shown in 9. Interesting results of this simulation are that the amount of force necessary to increase the length of a polymer is directly related to the temperature of the polymer. It can be seen that there are iso-lines of a constant length which show a linear dependence between force and temperature. What is unexpected about this plot is that all the iso-lines seem to converge in a single point. This point however, has a negative temperature, so it is not possible to run a physically relevant simulation to approach this point.

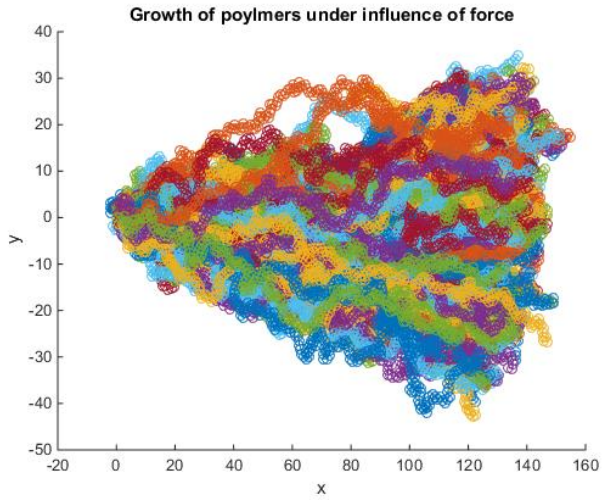

Fig. 6. Plot of a family of polymers growing with a force in the positive x direction applied to them. It can be seen clearly that the polymers give a preference to growth in the positive x direction over any other direction. The end to end distance therefore increases significantly with increasing force.

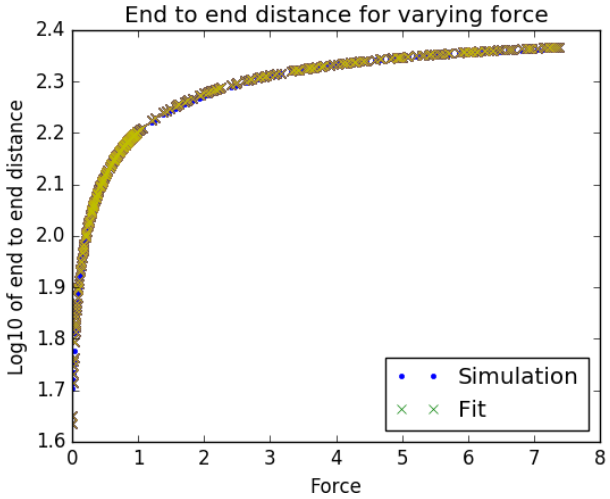

Fig. 7. End to end distance of the polymer population with varying force. The end to end distance of the 250th bead of the polymers is shown. This function seems to asymptotically approach the log10(250) line which makes sense as the polymer would grow to a length of 250 if we would apply an infinite force on the polymer. A fit has been plotted to the function which is also shown in this figure.

5. Conclusions

In this project we simulated the thermodynamic equilibrium of 2D polymers under different circumstances using the prune-enriched Rosenbluth method. We ran simulations for polymers growing up to a length of 250 beads. We validated our model by comparing the end to end distance of the polymer growth with the self-avoiding walk relation. After validating our model, we ran simulations for varying temperatures and varying forces. Low temperatures resulted in a curled up polymer because the Lennard-Jones attractive force is the most significant effect on

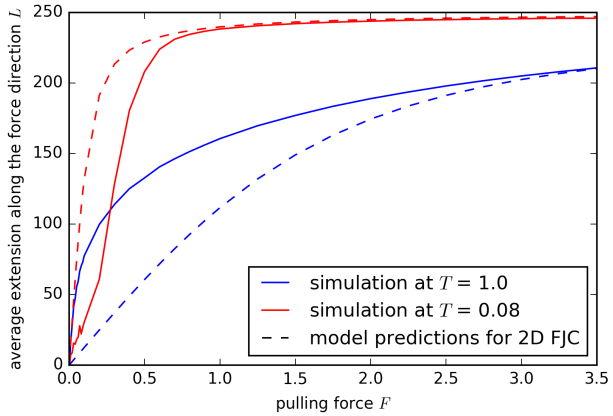

Fig. 8. Simulated force-extension curves for polymers in the high and the low temperature regime, compared with the theoretical force-extension curves for 2D freely-jointed chains as obtained in (5)

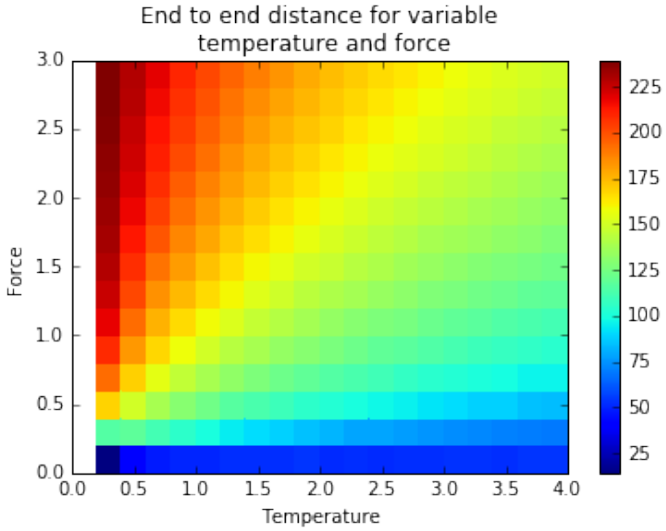

Fig. 9. End to end distance of polymers with a size of 250 beads for varying temperature and force. What is interesting in the image is that there are lines of polymers with the same length for different temperatures and forces. These lines seem to all go to the position. However, this position has a negative temperature.

the polymers. Increasing the temperature resulted in an increase in polymer end to end distance up to a temperature after which the end to end distance stabilized. At this point the thermal forces are dominant and the attractive Lennard-Jones forces becomes negligible. Therefore the growth of the polymer resembles a self-avoiding random walk simulation. Increasing the force resulted in an increase in end to end distance which approaches the asymptote of 250 which is the amount of beads in the polymer. The behaviour of the end to end distance was determined to be exponentially dependent on the polymer force. Comparison with a 2D freely-jointed chain model once again highlighted the repulsive LJ interaction at high temperatures and attractive LJ interaction at low temperatures. Finally a simulation was done where both the force and the temperature were varied. This simulation shows that for a higher temperature the force has less effect on the end to end distance of the polymer than at a lower temperature.

References

[1] Thijssen, J.M. (2007). Computational Physics. Cambridge University Press.
[2] Rosenbluth, M. N., Rosenbluth, A. W. (1955). Monte Carlo calculation of the average extension of molecular chains. The Journal of Chemical Physics, 23(2), 356-359.
[3] Grassberger, P. (1997). Pruned-enriched Rosenbluth method: Simulations of theta polymers of chain length up to 1 000 000. Physical Review E, 56(3), 3682.
[4] Baumgärtner, A. (1980). Statics and dynamics of the freely jointed polymer chain with Lennard-Jones interaction. The Journal of Chemical Physics, 72(2), 871-879.
[5] Iliafar, S., Vezenov, D., Jagota, A. (2013). Stretching of a Freely Jointed Chain in Two-Dimensions. arXiv preprint arXiv:1305.5951. ISO 690