# A Geometric Embedding Algorithm for Efficiently Generating Semiflexible Chains in the Molten State

**M. Kröger[1], M. Müller[2], and J. Nievergelt[2]**

**Abstract:** We present a novel method for generating starting polymer structures for molecular simulations in the dense phase. The work describes the ingredients of an algorithm for the creation of large, dense or diluted amorphous polymeric systems close to equilibrium and provides measures for its quality. The model systems are made of semiflexible (wormlike) repulsive multibead chains. The key feature of the method is its efficiency, in particular for large systems, while approaching given local and global chain characteristics. Its output has been proven to serve as an excellent basis for subsequent off-lattice molecular dynamics computer simulation. By combining chain growing with an iterative relaxation technique we remove overlaps of monomers. The computing time is linear in the number of beads and independent of chain length. The method succeeds in generating large and dense (bulky and confined) systems of up to $100,000$ beads in less than an hour on todays workstations.

**keyword:** semiflexible polymers, equilibrium structure, generation, benchmark, algorithm

## 1 Introduction

Computer simulations provide a powerful tool to investigate the microscopic origins of macroscopically observable transport phenomena. Monte Carlo type simulations are successfully used in the determination of conformational properties and phase diagrams of polymers [Binder and Ciccotti (1996)], molecular dynamics (MD) simulations are used to analyze the dynamics of polymeric systems and their response to external forces [Kremer et al. (1995), Allen and Tildesley (1987), Heermann (1988), Kröger (1998)]. Based on model fluids which are generated by our geometric embedding algorithm we use a MD simulation to provide information about the microstructural relaxation dynamics towards the equilibrium state, which - by definition - cannot be generated by a sample generator. In order to be feasible particulate simulations have to be based on simple, but not too simple, models [McLeish (2002)]. For modern perspectives on the variety of methods and a vision for future developments in multiscale simulations for nano- and micro-mechanics of materials see, e.g., [Ghoniem and Cho (2002), Srivastava and Atluri (2002)]. Recent results have confirmed that simulations can be used to predict rheological behavior of real polymers, even when the simulation model does not contain specific chemical details. This finding is not surprising in view that polymer melts possess certain 'universal' properties, cf. [de Gennes (1979)]. Although polymers may differ in flexibility or length, an appropriate reference parameter exists in each case that indicates the polymers are indeed similar in their inherent form. Because flow behavior is related to the polymer's inherent structure, qualitative rheological properties of polymer melts are largely independent of the chemical structure of the monomer repeat units. To visualize the features of the code we model a polymer melt in simple form as a mixtures of beads (soft balls), with $N$ beads per chain whose connectivity is realized by (nonlinear) springs (along the chains contours). In this flexible chain model a bead represents a number of chemical monomers.

A main difficulty when attempting to compute the statics and dynamics of dense model polymeric systems in equilibrium or even nonequilibrium situations by molecular/Brownian dynamics computer simulations is to generate and pre-equilibrate homogeneous samples by which these simulations can and should be started off, see [Kotelyanskii (1997), Brown et al (1994), Akkermans et al. (1998), Paul et al. (1997), Neyertz et al. (1996), Lin et al. (1996), Johnson (1996), Kröger (1999)]. This ini-

[1] Institute of Theoretical Physics, University of California Santa Barbara, USA, and Polymer Physics, ETH Zürich, Switzerland, and Institute of Theoretical Physics, TU Berlin, Germany. Email: editors@AR.ethz.ch

[2] Department of Computer Science, Institute of Theoretical Computer Science, ETH Zürich, Switzerland

tial phase may occupy up to the half of the overall CPU time needed to analyze the system properties. One of the reasons is the particularly small integration time step during this 'fluctuating' phase, another one is the enormous increase of characteristic relaxation time of polymer chains with chain length (powers of 2 in dilute solution and 3.5 in melts are common). Hence it seems to be of help to distribute an efficient method to create polymeric samples with specific characteristics, which, for any choice of the break off criterion, represent as close as possible an equilibrated sample, i.e., a sample with not only the 'correct' local but also realistic global features. For the novel method the break off criterion is the minimum distance between center coordinates of nonbonded monomers. The minimum distance criterion has the feature that the maximum force (amplitude) acting between pairs of particles is thereby fixed for the sample, after being created. The force amplitudes, and more precisely the Einstein frequency, gives an accurate estimate for the integration time step of a subsequent molecular or Brownian dynamics run [Allen and Tildesley (1987), Heermann (1988)]. For the homogeneous system this procedure is comparable with a minimum energy criterion.

Up to the authors knowledge the literature provides two comparable algorithms, which are very efficient for small system sizes and generate amorphous polymers both in the melt and glassy states. i) The method presented in [McKechnie et al. (1992)] possesses a three-stage structure. The basic chain topology was obtained by a simple self-avoiding random walk using the part that restraining bonds, bond angles, and torsions of the Hamiltonian but without any van der Waals interactions except those between sites on the same chain separated by three others (i.e., the pentane effect). Following chain growth, the excluded volume effects are introduced and the whole structure is relaxed by a molecular dynamics using the full Hamiltonian. ii) The algorithm `GenPol` provided in [Kröger (1999)] exhibits similarities, but the system parameters such as the integration time step are coupled to the conformation of the system in order to speed up the relaxation process. During a dynamical two-step process of sample creation the initially (Monte Carlo step 1) predicted global characteristics of the molecular conformations remain quite unaffected (during molecular dynamics step 2) and the potential energy and the entropy production are relaxing towards their minima. The potentials, the distribution of bond lengths, the integration time step and temperature are smoothly controlled during the creation/relaxation process until they finally approach their prescribed values. The quality of their algorithm is by its nature independent of concentration, system size or degree of polymerization; the CPU speed is quite independent of the latter quantity and linear in the system size. Chains tethered to a surface (dry polymer brushes) can be generated as well. The authors provided a benchmark table, whose entries will be compared with the ones produced by the novel algorithm for same parameters. Chain generation methods have been used for applications and also been extended, see, e.g., [Akkermans et al. (1998), Paul et al. (1997), Neyertz et al. (1996), Lin et al. (1996), Johnson (1996), Kröger and Voigt (1994), Kröger (2001), Neyertz and Brown (1996), Brown et al. (1996), Widmann and Suter (1995), Yang et al. (1999)].

In contrast to the techniques described in [McKechnie et al. (1992), Kröger (1999)] our new algorithm '`PolyGrow`' constructs the initial system of polymer chains by adding monomer beads one by one to a primarily empty box. This method is termed *chain growing* in the literature [Meirovitch (1983), Theodorou and Suter (1985)]. The new segments are added according to a choice criterion for achieving the target chain properties. At early stages of the chain growing process, the box is sparsely populated allowing the chains to grow in any direction. However, at later stages, the remaining monomers need to be fitted exactly into gaps left over by early stages. This major problem of the chain growing technique usually leads to asymmetric structures and to large sphere overlaps in dense packings. We solve this problem by combining chain growing with an iterative relaxation technique to remove overlaps of monomers. Since the relaxation is applied only locally, its time complexity is independent of the size of the system. Thus, the time complexity of the hybrid algorithm remains the same as the one of pure chain growing. The computing time is linear in the number of beads for the systems we investigate. `PolyGrow` succeeds in generating large and dense systems of up to $100,000$ beads in less than an hour on todays workstations.

The manuscript is organized as follows: Sec. 2 specifies the potentials for the polymer chain model under consideration and it describes the method to analyze its dynamics. Sec. 3 is devoted to the description of the algorithm to generate starting structures for dense polymers under various conditions. The speed and reliability of the algorithm are investigated in detail in Sec. 4.1 and Sec. 4.2, respectively.

## 2 The coarse grained polymer chain model

In the present context the 'wormlike chain model' is a combination of the model presented in [Kratky and Porod (1949)] and the finitely extendable nonlinear elastic force (FENE) model of [Warner (1972)]. The algorithm to be presented will still be useful if the precise form of the repulsive and attractive potentials is to be changed.

Each linear, semiflexible polymer chain within the bulk of surrounding polymers consists of $M$ beads, whereby neighboring beads along the chain are connected by an anharmonic attractive FENE spring [Kremer and Grest (1990), Warner (1972)]. In order to prevent beads from overlapping, all pairs of beads (inter- and intramolecular pairs) interact with a repulsive Lennard-Jones potential [Weeks et al. (1971)]

$$U_{ij}^{\mathrm{LJ}} = \begin{cases} 4[(r_{ij}^*)^{-12} - (r_{ij}^*)^{-6} + 1/4] & \text{for } r_{ij}^* \le 2^{1/6} \\ 0 & \text{for } r_{ij}^* \ge 2^{1/6} \end{cases}, \tag{1}$$

where $\mathbf{r}_{ij}^* \equiv \mathbf{r}_j - \mathbf{r}_i$ denotes the vector between beads at space coordinates $\mathbf{r}_i$ (with $i = 1, \ldots, N$) and $r_{ij}^* \equiv |\mathbf{r}_{ij}^*|$. Here and in the following all quantities which are reduced to the usual LJ-units are denoted by an asterisk if otherwise ambiguities could arise. For beads which are nearest neighbors along the chain (for $N > 1$), an attractive potential (FENE-potential) is added, see e.g. [Kröger, Luap and Muller (1997)]:

$$U_{ij}^{\mathrm{FENE}} = \begin{cases} -0.5k^* R_0^2 \ln[1 - (r_{ij}^*/R_0)^2] & \text{for } r_{ij}^* \le R_0 \\ \infty & \text{for } r_{ij}^* \ge R_0 \end{cases}. \tag{2}$$

So far, these potentials set up a widely used model for flexible chains. Kratky and Porod introduced a semiflexibility by assuming an elastic bending energy for the chain

$$U^{\mathrm{bend}} = \frac{\kappa}{2} \int_0^L ds \, (\frac{\partial \mathbf{u}}{\partial s})^2 = \kappa a_0 \sum_{i=1}^{N-2} (1 - \frac{\mathbf{u}_i \cdot \mathbf{u}_{i+1}}{|\mathbf{u}_i \cdot \mathbf{u}_{i+1}|}), \tag{3}$$

with segment vectors $\mathbf{u}_i \equiv \mathbf{r}_{i+1} - \mathbf{r}_i$, temperature $T$, Boltzmann's constant $k_B$, bond length $a_0 \equiv \langle |\mathbf{u}| \rangle$ and a persistence length $l_p \equiv \kappa / k_B T$, see e.g. [Doi and Edwards (1996)].

The novel algorithm is also designed to generate polymer brushes – in addition to free polymeric chain systems – in order to compare with previous works. For the case of a brush one of the tails of each molecule is conveniently located in the (x-y) plane, and chains do not cross this plane. For these systems the only additional parameter is the surface concentration $\sigma$ (number of tails per area).

To summarize, the parameters of our wormlike (free) chain model are: the total number of beads $N$, the number of beads per chain $M$ or the number of chains $N/M$, the bond length $a_0$, the bending rigidity $\kappa$, the temperature $T$, the bead number concentration $\rho$, the maximum extensibility $R_0$ and spring coefficient $k$ of the FENE chain, and the surface concentration $\sigma$ for polymer brushes.

The wormlike chain model can be simulated by molecular dynamics (MD) starting from an equilibrated initial start configuration. A MD computer simulation is a method for solving a classical many-particle problem. Using the model potential as a basis, the discretized Newtonian equations of motions are solved iteratively. A polymers physical quantities are extracted within a central simulation cell that contains all the monomers. Since we are interested in bulk properties, this original cell with all its particles is repeated periodically in all directions of space, except for the case of brushes (see previous section). The interaction between two monomers is calculated for their spatially closest representant; which can be original or image particles in neighboring cells. Based on previous research [Kremer and Grest (1990), Kröger et al. (1993)], a short-ranged Lennard-Jones, Eq. 1, upon a long-ranged intermolecular potential is used, which - for dense systems - covers the main physics and is advantageous in terms of the required calculation time.

## 3 The `PolyGrow` Algorithm

`PolyGrow` handles the following problem: Given the model parameters $d < d_{\max}$, $a_0$, $\kappa$, the number of beads $N$, the number of chains $N/M$ and the number of beads per cube unit $\rho$, find coordinates for the centers of all $N$ beads such that the spheres do not overlap and that a given distribution of bond angles is obtained. The quantity $d_{\max}$ corresponds to the actual maximum packing density at given bead density $\rho$.

To generate appropriate starting structures of wormlike polymers described in the previous section we hence process a simplified geometric model. The polymer is modeled by a sequence of beads with fixed diameter $d$, which are connected by bonds of fixed length $a_0$. The target initial distribution of bond angles ($\theta$), according to Eq. 3 is given by: $\mathrm{prob}(\theta) \propto \exp[\kappa(1-\cos\theta)]$.

### 3.1 Chain growing

The main loop of `PolyGrow` adds $n$ chains of $m$ beads to a primarily empty periodic box. This is done by repeatedly calling a procedure `AddBead`. If `AddBead` fails, the actual chain is removed and regrown at a new location. The pseudo code for `AddBead` reads

```
procedure AddBead:Boolean;
begin
   forall p ∈ possible positions do
      if LookAhead(p, searchDepth) then
         attach new bead at position p;
         return true;
      endif;
   endfor;
   forall p ∈ possible positions do
      attach new bead at position p;
      if Relax() then return true;
      remove new bead;
   endfor;
   return false;
end;
```

`AddBead` evaluates a discrete set of possible positions for the new bead. This step depends on the particular model of the polymer system e.g. the parameters $a_0$ and $\kappa$. Then, the positions are tested by the procedure `LookAhead` in random order. `LookAhead` performs a backtrack search of constant depth. It checks whether the chain can be elongated via the new position by a certain number of beads without overlaps. This step is necessary because the chains are not completely flexible. Their local flexibility is biased by the bending rigidity $\kappa$:

```
procedure LookAhead(p, depth):Boolean;
begin
   if depth ≤ 0 then return true;
   forall $p_{succ}$ ∈ possible successors of p do
      if $p_{succ}$ does not overlap then
         if LookAhead($p_{succ}$, depth-1)
            then return true;
      endif;
   endfor;
   return false;
end;
```

If inside the first loop of `AddBead` a position **p** is found that can be extended by at least `depth` beads without overlaps, a new bead is attached at this location. Otherwise another strategy is applied: The second loop tests all possible positions in the order of their expected quality. The quality is measured by the overlap of **p** or by the number of beads the chain can grow via **p** without overlap.

The new bead is attached at position **p** although it may generate overlaps with existing beads. If the new bead does not overlap with any other beads, `AddBead` returns successfully. Otherwise, the newly produced overlaps need to be removed. This is not a trivial task, since the colliding beads belong to earlier generated chains. If these beads are moved, adjacent bond lengths as well as bond angles will change. However, neither bond lengths, nor bond angle statistics should be disturbed during the collision correction. The only way to modify the positions of beads inside a chain while keeping bond lengths and bond angles fixed, is a simultaneous (concerted) rotation of at least seven torsion angles. This move is described in detail in [Dodd et al. (1993)] where a complex procedure to evaluate the values of the seven torsion angles is presented. This procedure needs to be performed simultaneously on all the chains, contributing to the collisions of the newly positioned bead.

### 3.2 Local relaxation

Fortunately there is a way to avoid the complex computations of concerted rotations. Since the set of beads that need to be moved form a rather small and local system of particles, a simple relaxation algorithm is suitable to solve the local packing problem. The algorithm `Relax` we use is simple and general in the sense that it can handle problems of the following form: Given a set of

**Table 1** : Benchmarks obtained with GenPol (R10000 at 195 MHz) and PolyGrow (Ultra-Sparc-II at 248 MHz) for the generation of monodisperse polymer systems of different size, bulk density and different choices for the final minimum distance between monomers. The specified samples S1-S5 are discussed in more detail in this work.

| | total | chain | chains | bead | surface | minimum | GenPol | PolyGrow |
|---|---|---|---|---|---|---|---|---|
| | beads | length | | density | density | distance | time | time |
| sample | $N$ | $M$ | $N/M$ | $\rho$ | $\sigma$ | d | (s) | (s) |
| — | 1000 | 1000 | 1 | 0.8 | — | 0.8 | 22 | 8 |
| S1 | 1000 | 1000 | 1 | 0.9 | — | 0.9 | – | 16 |
| — | 1000 | 1000 | 1 | 1.0 | — | 0.8 | 27 | 12 |
| S2 | 1000 | 100 | 10 | 0.9 | — | 0.9 | 24 | 26 |
| — | 1000 | 100 | 10 | 1.0 | — | 0.9 | 25 | 57 |
| — | 5000 | 100 | 50 | 0.8 | — | 0.8 | 130 | 39 |
| — | 10000 | 100 | 100 | 0.8 | — | 0.8 | 340 | 82 |
| S5 | 30000 | 50 | 600 | 0.8 | — | 0.9 | – | 287 |
| — | 100000 | 100 | 1000 | 0.8 | — | 0.8 | 8050 | 959 |
| S3 | 5000 | 100 | 50 | 0.4 | 0.2 | 0.8 | 69 | 23 |
| S4 | 5000 | 100 | 50 | 0.8 | 0.4 | 0.8 | 240 | 206 |

### 3.3 The Complexity of PolyGrow

To analyze the time complexity of PolyGrow, we split the computing time into the time used by AddBead to add a new bead to the system and the number of times AddBead is called.

The time used by AddBead is independent of the problem size. The number of possible positions for the new bead is constant. For each position, a search of constant depth is started. If there is no space for the new bead, Relax is called at most once for each new position. The number of particles Relax moves is independent of the problem size. Therefore, the total time used to add one bead is independent of the total number of particles in the system.

Since AddBead can fail, it is difficult to predict how many times it is called during the packing process. If $N$ is the number of beads, AddBead is called exactly $N$ times in case it never fails. This leads to linear time complexity $O(N)$ in the best case.

As our experiments show, PolyGrow reaches linear time complexity for realistic systems of wormlike polymer chains (see section 4.1).

### 3.4 Polymer brushes and z-alignment

PolyGrow allows to generate special kinds of polymer systems such as polymer brushes and z-aligned chains. These additional constraints only affect the part of the code inside AddBead where the set of possible positions is produced. For polymer brushes, the first bead is fixed in the $z = 0$ - plane. Additional distance constraints prevent the chains from crossing the top and bottom planes of the periodic box. Z-alignment is implemented by favouring new positions along the z-axis.

## 4 Results

### 4.1 Time needed to generate pre-equilibrated samples

To analyze the time complexity of PolyGrow we adopted the default system parameters from [Kröger (1999)]: systems size $N = 1,000$, chain length $M = 100$, density $\rho = 0.9$, diameter (break off criterion) $d = 0.9$ and stiffness $\kappa = 4$ and varied the three relevant parameters: $N, \rho$ and $M$.

Figure 1 depicts the time used by PolyGrow to pack systems of various sizes $N$. We compare our results with the performance of GenPol [Kröger (1999)], the fastest method to generate wormlike amorphous polymer systems at present. PolyGrow shows a linear behavior while the time used by GenPol increases slightly steeper. The squares show the time used by GenPol, the triangles represent average values over 5 runs of PolyGrow. The two lines above and below the triangles show the standard deviation. PolyGrow is between two and five times faster

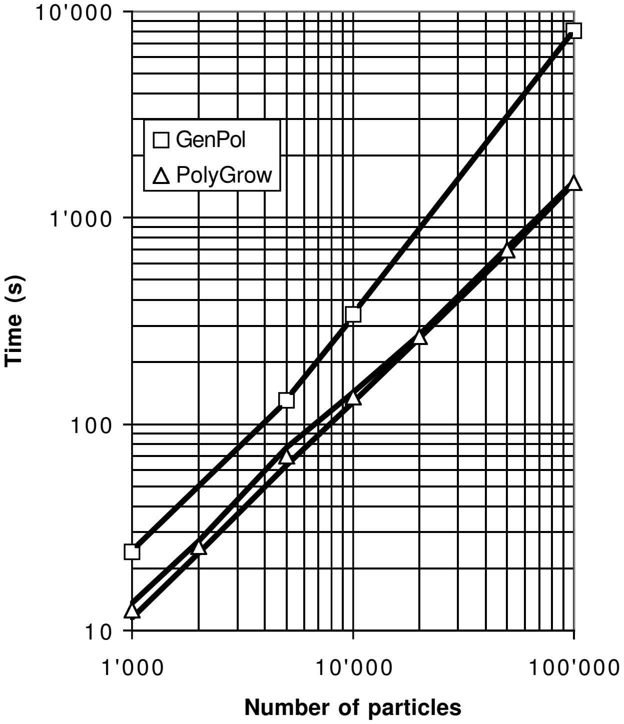

**Figure 1** : Computing time of `PolyGrow` (MIPS R5000 at 180 MHz) versus system size $N$ compared to results obtained with `GenPol` (MIPS R10000 at 195 MHz). System parameters: $N = 1,000 \ldots 100,000$, $M = 100$, $\rho = 0.8$, $d = 0.8$ and $\kappa = 4$.

than `GenPol` depending on the system size.

The dependence of the execution time on the systems density ρ is nonlinear. Above a certain value for ρ the problem becomes unsolvable, there is no way to pack the chains without overlapping spheres. Up to the conventional melt density of 0.9 particles per cube unit, the execution time does not increase dramatically, as Fig. 2 shows. However, `PolyGrow` is unsuitable for densities above 1.0. For very high densities, `GenPol` is more efficient. It is able to pack systems up to 1.5 particles per cube unit.

The final parameter to be analyzed is the chain length $M$. As depicted in Fig. 3, the execution time of `PolyGrow` is independent of the lengths of the chains. The standard deviations over 20 runs are much higher than the differences in the average values. The task to finish a chain and to begin an new one during the chain growing process does neither take too much time, nor does this additional degree of freedom simplify the problem significantly.

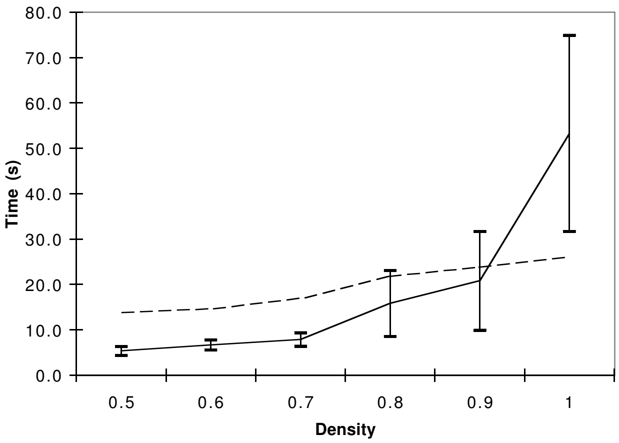

**Figure 2** : Computing time of `PolyGrow` versus system density (Ultra-Sparc-II at 248 MHz). System parameters: $N = 1,000$, $M = 100$, $\rho = 0.5 \ldots 1.0$, $d = 0.9$ and $\kappa = 4$. The dashed line represents reference results obtained with `GenPol` [Kröger (1999)].

### *4.2 The quality of generated structures*

In order to check the quality of the algorithm and the reliability of the generated samples we will show some dynamical quantities, i.e. relaxation behavior obtained by subsequent conventional molecular dynamics (MD) simulation [Kröger et al. (1993), Kröger, Luap and Muller (1997)].

All the samples listed in Tab. 1 were generated by our geometric embedding algorithm and used as starting configuration for the MD. In addition to the listed parameters the MD requires to specify the interaction potential between neighboring beads within chains and the intermolecular interaction potentials as well as the integration time step. Newton's equation of motion were integrated using the velocity Verlet algorithm, neighbor lists, layered link cells, see e.g. [Allen and Tildesley (1987), Rapaport (1991), Kremer and Grest (1990), Kröger et al. (1993)] and Lees-Edwards boundary conditions [Allen and Tildesley (1987)]. with a time step of $\triangle t = 1/200$. With the choice for the finite extensibility of the FENE-spring $R_0 = 1.5$ and $k^* = 30$ we follow previous investigations [Kremer and Grest (1990), Kröger et al. (1993), Kröger (2001)].

Next, we will analyze (and give the definitions for) quantities which have been extracted from the bead trajectories during the coarse of the 'computer' relaxation exper-

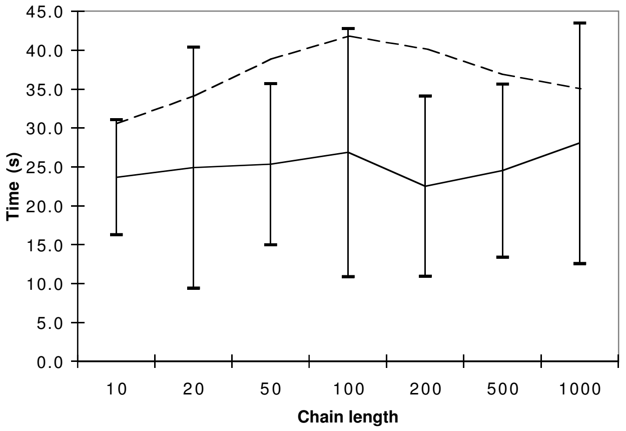

**Figure 3** : Computing time of `PolyGrow` versus chain length. (Ultra-Sparc-II at 248 MHz). System parameters: $N = 1,000$, $M = 10\ldots1000$, $\rho = 0.9$, $d = 0.9$ and $\kappa = 4$. As in Fig. 8 the dashed line represents reference results obtained with `GenPol`.

iments.

*Structure factor and radius of gyration*

A structural quantity, which is directly related to the conformation of polymers, and which can be measured experimentally, e.g., by small angle Neutron scattering methods, is the static structure factor of individual chains $S_{\rm sc}(\mathbf{k})$. The structure factor provides information about degree of the flow alignment on a local and global length scale, from the segment length to the length of the extended polymer chain. From the MD computer simulation this quantity can be extracted by a fast Fourier transform of the pair correlation function, or explicitly from the bead coordinates [Kröger et al. (1993)]

$$S_{\rm sc}(\mathbf{k}) = \frac{1}{N}\sum_{\alpha}^{N/M}\left\langle \mid \sum_{i}^{M} \exp^{i\mathbf{k}\cdot\mathbf{r}_i^\alpha} \mid^2 \right\rangle, \tag{4}$$

with $\mathbf{r}_i^\alpha$ being the position of the $i$th bead of molecule $\alpha$; $N$ is the total number of beads, $N/M$ is the number of chains. The structure factor of individual chains is related to the conformational properties of the polymers, e.g., the squared radius of gyration $R_{\rm g}^2 = \sum_{i>j}^{M-1}(\mathbf{r}_i - \mathbf{r}_j)^2/M^2$ is pursued from the initial slope by

$$M S_{\rm sc}(k)^{-1} = (1 - k^2\frac{R_g^2}{3})^{-1} \approx 1 + k^2\frac{R_g^2}{3} \quad \text{for } kR_G \ll 1. \tag{5}$$

This expression is obtained by a Taylor series expansion of $S_{\rm sc}$ in Eq. 4 around $\mathbf{k} = 0$. It provides a direct measure of the length $M$ of polymer chains.

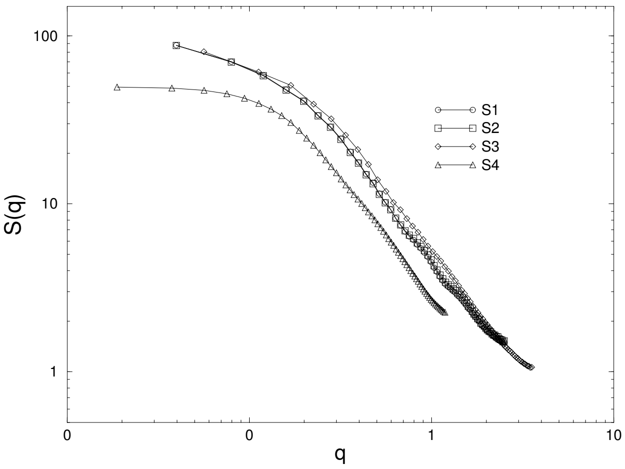

**Figure 4** : Single chain structure factors $S_{\rm sc}(q)$ vs wave number $q$ as a measure for the be ad-bead correlation on all length scales (in reciprocal space) for samples S1-S4 (see Eq. 4).

The structure factors for the equilibrated samples S1-S4 are given in Fig. 4 and the quality of a specific sample is demonstrated by Fig. 5. The chains are 'pre-equilibrated' on all lengths scales directly after their generation by the novel algorithm. This is quite remarkable in view of previous studies, see [Kröger (1999)]. In addition to the structure factors we extracted the relaxation of the radius of gyrations $R_G$ towards its equilibrium value, see Fig. 6 for all samples.

*Order parameter*

A quantity characterizing the systems anisotropy is called order parameter. For polymer brushes there is a single distinguished, the one which is perpendicular to the anchoring plane, which we have chosen to be $x$-$y$-direction. The order parameter is defined as the average $\langle P_2(\mathbf{u}\cdot\mathbf{e}_z)\rangle$ over all segment vectors $\mathbf{u}$, where $\mathbf{e}_z$ (in our case) is directed along the $z$-axis and $P_2$ is the second Legendre polynomial. For bulk samples composed of free flexible chains the order parameter has to vanish, as soon as the chains become less flexible, local or even global ordering of chains may take place, resulting from steric interactions, and an isotropic to nematic transition occurs. This leads to large fluctuations in the

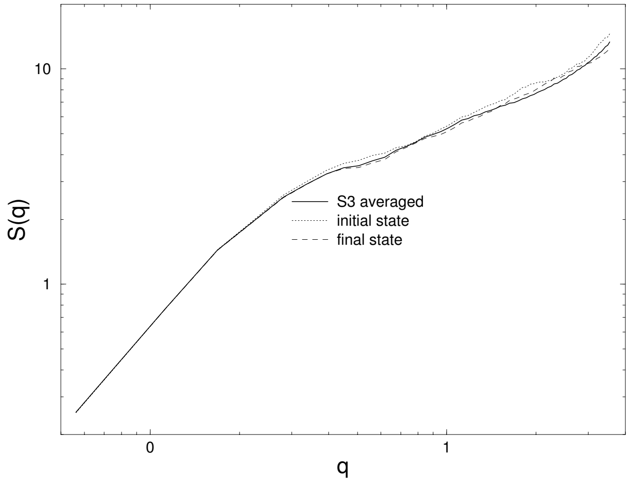

**Figure 5** : Single chain structure factors $q^2 S_{\rm sc}(q)$ (Kratky-representation) vs wave number $q$ for sample S3. a) time average, b) initial configuration, c) final configuration. The small differences between the three curves actually prove the good quality of the generated molecules on all length scales.

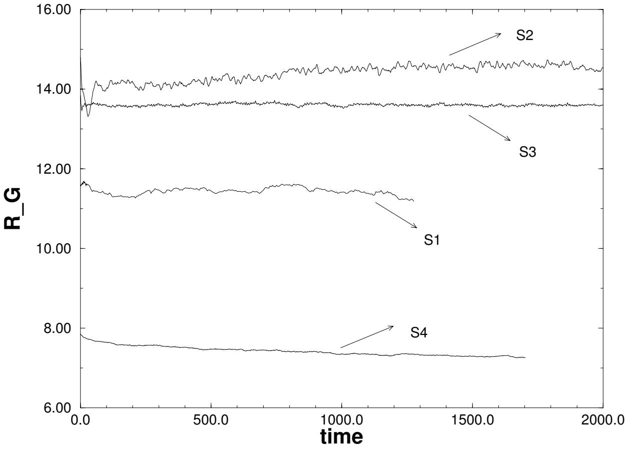

**Figure 6** : Radius of gyration as a measure for the global stiffness of chains vs time for samples S1-S4 (see Sec. 4).

order parameter. The order parameter for the polymer brushes S2, S3 is shown in Fig. 7. Both samples possess the same anisotropy at the startup of the equilibration phase (time $t \equiv 0$, the first extraction was made at $t = \triangle t$), but relax to different values, in accord with the different bead densities which do not enter our sample generator. With increasing densities (bulk and surface) the chains increase their alignment perpendicular to the anchoring plane. The observed relaxation time is of the order of magnitude of the orientational relaxation time of chains, and thus, as fast as possible.

*Persistence length*

According to Eq. 3 the persistence length, a measure for the local stiffness, is obtained from

$$l_p = \frac{1}{2(M-2)} \sum_{i=1}^{M-2} \mathbf{u}_i \cdot \mathbf{u}_{i+1}. \qquad (6)$$

with segment vectors $\mathbf{u}_i$ $(i = 1,..,M-1)$. The relaxation of persistence lengths for all samples towards their equilibrium values is reported in Fig. 8. Again, the deviation between equilibrium values and initial values (at time=0) is small, as required for an algorithm which does not only fulfill the constraint of minimum distance, but which does also account for the quality of the conformational statistics.

A good indicator for the quality of a generated sample is its Einstein frequency $\omega_{\rm E} \propto \langle \mathbf{F}_i^2 \rangle$, and, in particular, the relaxation time of this quantity. Here, $\mathbf{F}_i$ denotes the total force acting on bead $i$ at a given time. Fast relaxation of $\omega_{\rm E}$ is crucial for choosing a large time step, i.e., for a large ratio: physical time/computing time. Figure 9 reports about the behavior for polymer brushes (samples S2, S3) - the nasty case -, where the relaxation time is ‘large’ (but still below a single time unit). Compared to the Einstein frequency in other nonequilibrium situations, this is more fast than expected and holds as an additional plus for the presented generator.

## 5 Conclusions

This work provides the structural elements of a method to generate macromolecular fluids close to equilibrium, which is basically needed and a difficult task for all types of molecular dynamics computer simulations dealing with polymers in the glassy, molten or highly concentrated states. The problem is geometrical in nature and does not occur for short molecules. The work includes quantitative measures of the quality (local and global entities, molecular structure factors) of the method as well as a benchmark table in order to make the comparison with earlier works transparent.

We combined a constructive chain growing process with an iterative technique to remove local overlaps of beads.

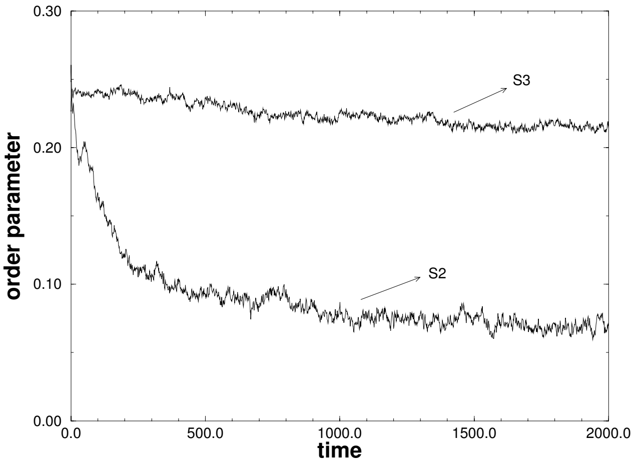

**Figure 7** : Transient behavior of the order parameter for the polymer brushes, samples S2-S3. The order parameter measures the degree of alignment in direction perpendicular to the anchoring plane, which is the $x-y-$ direction, and the order parameter is defined as the average $\langle P_2(\mathbf{u}\cdot\mathbf{e}_z)\rangle$ over all segment vectors $\mathbf{u}$, where $\mathbf{e}_z$ is directed along the $z$-axis. For samples S1, S4 (free chains) the order parameter approximately vanishes, which is not shown here, but is reflecting the quality of the sample generator as well.

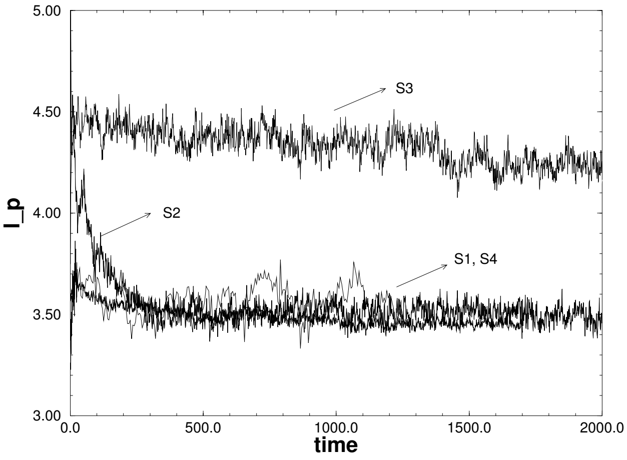

**Figure 8** : Persistence length as a measure for the local stiffness of chains vs time for samples S1-S4 (see Sec. 4, Tab. 1).

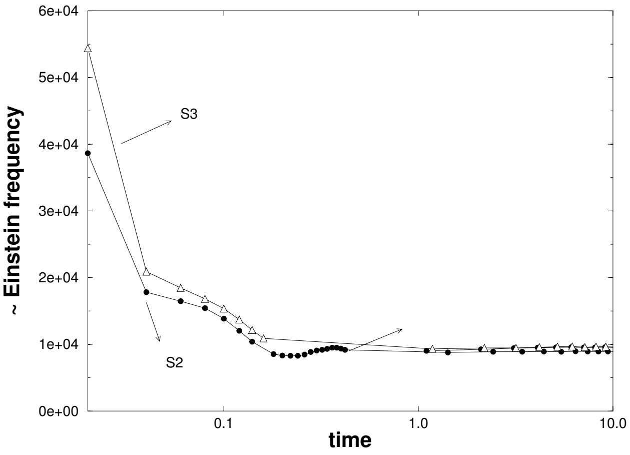

**Figure 9** : The sums of squared interaction forces for samples S2 and S3 (polymer brushes, see Tab. 1), respectively, which are proportional to the Einstein frequencies of the particular systems. For these samples the relaxation of this quantity towards its mean value is slow compared to that of the other samples containing unconstrained polymers.

This hybrid algorithm permits to tackle the problem of packing large systems of wormlike polymers (up to $100,000$ beads) into periodic boxes five times faster than the best of today's methods. Chain growing is very efficient but it may run into troubles during the packing process. On the other hand, relaxation is much more expensive but it is needed only locally in the environment of the newly placed bead. The combination of the both processes turns out to be a suitable way of constructing large and dense polymer systems. The method was clearly motivated by previous works [McKechnie et al. (1992), Kröger (1999)].

The intra- and intermolecular potentials used in the simulations define the microscopic model of the polymer fluid. The coarse grained model used here is able to reproduce properties of real substances as well as providing complete information on the microscopic molecular motions. Recent results have shown that simulations can be used to predict rheological behavior of real polymer melts, even if the simulation model does not contain specific chemical details [Hess (1987), Gao and Weiner (1992), Kröger (2001)] in accordance with predictions on self-similarity and universality in polymer systems [de Gennes (1979)].

(polymer brushes,

From a simulation of a polymer system one correlates the polymer microstructure (i.e. the form and the orientation of the chains and entanglements between chains) with its mechanical behavior. The simulation is conducted to test and improve mesoscopic theories such as those that describe orientation and distribution functions of polymers [Bird et al. (1987), Öttinger (1996)]. Only a portion of the available microscopic information is reflected in these theories. A further advantage of the simulation is its capability of focusing on measuring a particular quantity, such as specific contributions to the stress tensor, entanglements between polymer chains or the scattering diagram of individual polymers.

## References

**Akkermans, R. L. C.; Toxvaerd, S.; Briels, W. J.** (1998): Molecular dynamics of polymer growth. *J. Chem. Phys.*, vol. 109, pp. 2929-2940.

**Allen, M. P.; Tildesley, D. J.** (1987): *Computer simulation of liquids*, Clarendon, Oxford.

**Binder, K.; Ciccotti G., eds.** (1996): *Monte Carlo and molecular dynamics of condensed matter systems*, IPS Conf. Proceedings **49**, Nuovo Cimento, Bologna.

**Bird, R. B.; Hassager, O.; Armstrong, R. C.; Curtiss, C. F.** (1987): *Dynamics of Polymeric Liquids*, Vol. 2, Wiley & Sons, NY.

**Brown, D.; Clarke, J. H. R.; Okuda, M.; Yamazaki, T.** (1994): The preparation of polymer melt samples for computer simulation studies. *J. Chem. Phys.*, vol. 100, pp. 6011-6018.

**Brown, D.; Clarke, J. H. R.; Okuda, M.; Yamazaki, T.** (1996): A large scale molecular dynamics study of chain configurations in the n=100 alkane melt. *J. Chem. Phys.*, vol. 104, pp. 2078-2082.

**de Gennes, P.G.** (1979): *Scaling concepts in polymer physics*, Cornell Univ. Press, Ithaca, NY.

**Doi, M.; Edwards, S. F.** (1996): *The theory of polymer dynamics*, Clarendon, Oxford.

**Dodd, L. R.; Boone, T. D.; Theodorou, D. N.** (1993): A concerted rotation algorithm for atomistic Monte Carlo simulation of polymer melts and glasses. *Mol. Phys.*, vol. 78, pp. 961.

**Gao, J.; Weiner, J. H.** (1992): Stress relaxation in a polymer melt of freely-rotating chains. *J. Chem. Phys.*, vol. 97, pp. 8698.

**Ghoniem, Nasr. M.; Cho, Kyeongjae** (2002): The Emerging Role of Multiscale Modeling in Nano- and Micro-mechanics of Materials. *CMES: Computer Modeling in Engineering & Sciences*, vol. 3:2, pp. 147-174.

**Heermann, D. W.** (1988): *Computer simulation methods in theoretical physics* Springer, Berlin.

**Hess, S.** (1987) Rheological properties via nonequilibrium molecular dynamics: From simple towards polymeric liquids. *J. Non-Newt. Fluid Mech.*, vol. 23, pp. 305.

**Johnson, J. K.** (1996): Perturbation theory and computer simulations for linear and ring model polymers. *J. Chem. Phys.*, vol. 104, pp. 1729-1742.

**Kotelyanskii, M.** (1997): Simulation methods for modeling amorphous polymers. *Trends Polym. Sci.*, vol. 5, pp. 192-198.

**Kratky, O.; Porod, G.** (1949): Röntgenuntersuchung gelöster Fadenmoleküle. *Recl. Trav. Chim. Pay B*, vol. 68, pp. 1106-1122.

**Kremer, K.; Grest, G. S.** (1990): Dynamics of entangled linear polymer melts: a molecular-dynamics simulation. *J. Chem. Phys.*, vol. 92, pp. 5057-5086.

**Kremer, K.; Grest, G. S.** (1995): Computer simulations in polymer physics, *Physics World*, vol. 3, pp. 39.

**Kröger, M.; Loose, W.; Hess, S.** (1993): Structural changes and rheology of polymer melts via nonequilibrium molecular dynamics. *J. Rheol.*, vol. 37, pp. 1057.

**Kröger, M.; Voigt, H.** (1994): On a quantity describing the degree of entanglement in linear polymer systems. *Macromol. Theor. Simul.*, vol. 3, pp. 639-647.

**Kröger, M.** (1995): NEMD computer simulation of polymer melt rheology. *Appl. Rheol.*, vol. 5, pp. 66.

**Kröger, M.; Luap, C.; Muller, R.** (1997): Polymer melts under uniaxial elongational flow: stress-optical behavior from experiments and NEMD computer simulations. *Macromolecules*, vol. 30, pp. 526.

**Kröger, M.** (1998): Nonequilibrium molecular simulations of simple and complex fluids. *Curr. Opin. Coll. Interf. Sci.*, vol. 3, pp. 614-619.

**Kröger, M.** (1999): Efficient hybrid algorithm for the dynamic creation of semiflexible polymer solutions, brushes, melts and glasses. *Comput. Phys. Commun.*, vol. 118, pp. 278.

**Kröger, M.** (2001): Rheological evidence for a dynamical crossover in polymer melts via nonequilibrium molecular dynamics. *Phys. Rev. Lett.*, vol. 85, pp. 1128-

1131.

**Lin, B.; Boinske, P. T.; Halley, J. W.** (1996): A molecular dynamics model of the amorphous regions of polyethylene oxide. *J. Chem. Phys.*, vol. 105, pp. 1668-1681.

**McKechnie, J. I.; Brown, D. ; Clarke, J. H. R.** (1992): Methods of generating dense relaxed amorphous polymer samples for use in dynamic simulations. *Macromolecules*, vol. 25, pp. 1562.

**McLeish, T. C. B.** (2002): Tube theory of entangled polymer dynamics. *Adv. Phys.*, vol. 51, pp. 1379-1527.

**Meirovitch, H.** (1983): Computer Simulation of Self-avoiding Walks: Testing the Scanning Method. *J. Chem. Phys.*, vol. 79, pp. 502.

**Neyertz, S.; Brown, D.; Clarke, J. H. R.** (1996): The local energy approximation and the predictability of chain configurations in polymer melts. *J. Chem. Phys.*, vol. 105, pp. 2076-2088.

**Neyertz, S.; Brown, D.** (1996): Local structure and mobility of ions in polymer electrolytes: A molecular dynamics simulation study of the amorphous PEO(x)NaI system. *J. Chem. Phys.*, vol. 104, pp. 3797-3809.

**Öttinger, H. C.** (1996): *Stochastic Processes in Polymeric Fluids*, Springer, Berlin.

**Paul, W.; Smith, G. D.; Yoon, D. Y.** (1997): Static and dynamic properties of a n-C100H202 melt from molecular dynamics simulations. *Macromolecules*, vol. 30, pp. 7772-7780.

**Rapaport, D. C.** (1991): Multi-million particle molecular dynamics. I. Design considerations for vector processing. *Comput. Phys. Commun.*, vol. 62, pp. 198.

**Srivastava, D.; Atluri, S. N.** (2002): Computational Nanotechnology: A Current Perspective. *CMES: Computer Modeling in Engineering & Sciences*, vol. 3:5, pp. 531-538.

**Theodorou, D. N.; Suter, U. W.** (1985): Atomistic Modeling of Mechanical Properties of Polymeric Glasses. *Macromolecules*, vol. 19, pp. 139.

**Warner, H. R.** (1972): Kinetic theory and rheology of dilute suspensions of finitely extendable dumbbells. *Ind. Eng. Chem. Fund.*, vol. 11, pp. 379.

**Weeks, J. D.; Chandler, D.; Andersen, H. C.** (1971): Role of repulsive forces in determining the equilibrium structure of liquids. *J. Chem. Phys.*, vol. 54, pp. 5237.

**Widmann, A. H.; Suter, U. W.** (1995): Parallelization of a Monte-Carlo algorithm for the simulation of polymer melts *Comput. Phys. Commun.*, vol. 92, pp. 229-251.

**Yang, L.; Srolovitz, D. J.; Yee, A. F.** (1999): Molecular dynamics study of isobaric and isochoric glass transitions in a model amorphous polymer. *J. Chem. Phys.*, vol. 110, pp. 7058-7069.