

Pruned-enriched Rosenbluth method: Simulations of θ polymers of chain length up to 1 000 000

Peter Grassberger

HLRZ, Kernforschungsanlage Jülich, D-52425 Jülich, Germany

and Department of Theoretical Physics, University of Wuppertal, D-42097 Wuppertal, Germany

(Received 16 December 1996)

We present an algorithm for simulating flexible chain polymers. It combines the Rosenbluth-Rosenbluth method with recursive enrichment. Although it can be applied also in more general situations, it is most efficient for three-dimensional θ polymers on the simple-cubic lattice. There it allows high statistics simulations of chains of length up to $N=10^6$. For storage reasons, this is feasible only for polymers in a finite volume. For free θ polymers in infinite volume, we present very high statistics runs with $N=10\,000$. These simulations fully agree with previous simulations made by Hegger and Grassberger [J. Chem. Phys. **102**, 6681 (1995)] with a similar but less efficient algorithm, showing that logarithmic corrections to mean field behavior are much stronger than predicted by field theory. But the finite volume simulations show that the density inside a collapsed globule scales with the distance from the θ point as predicted by mean field theory, in contrast to claims in the work mentioned above. In addition to the simple-cubic lattice, we also studied two versions of the bond fluctuation model, but with much shorter chains. Finally, we show that our method can be applied also to off-lattice models, and illustrate this with simulations of a model studied in detail by Freire *et al.* [Macromolecules **19**, 452 (1986) and later work]. [S1063-651X(97)10308-7]

PACS number(s): 36.20.-r, 83.20.Jp

I. INTRODUCTION

Although chain polymers were among the first objects simulated on electronic computers, they still present a challenge both because of their importance in chemistry and biology, and because of the particular structure of the problem. The latter implies that straightforward algorithms [such as simple sampling (SS) [1]] are often inefficient, and that a host of methods have been proposed, all with their strengths and weaknesses [1]. Today it seems that the pivot algorithm is the most popular [2,3], since it is the fastest for relatively open (dilute) systems. But it is less easy to estimate entropies with the pivot algorithm, and it becomes inefficient in dense or constrained systems where most of the global moves (which make it fast in open and dilute systems) are forbidden.

In the present problem we will deal specifically with θ polymers, i.e., with self-avoiding walks (SAW's) with an additional nearest neighbor attractive interaction. On large scales it overrides the repulsion, so that the typical configuration changes then from an open "coil" to a dense "globule." In order to squeeze very long chains into finite computer memory, and in order to get rid of globular surface effects, we shall study the longest chains in a finite cube with periodic boundary conditions. For this system the pivot algorithm would be practically useless. But as we shall see, even if we consider open systems, our present algorithm is superior for accessible chain lengths ($\approx 10^4$)—in particular since it easily allows one to compute free energies.

Our results confirm that the θ -point collapse is a tricritical phenomenon with upper critical dimension $d_c=3$, whence the asymptotic behavior is mean-field-like [4]. But it also confirms recent claims [5] that corrections to this are much stronger than the logarithmic corrections predicted from the field theoretic renormalization group [6]. Moreover, we will show that this holds also for off-lattice and for a lattice with

very high coordination number, suggesting thus that the phenomenon is not a lattice artifact.

In the next section we will present general features of the algorithm. Specific aspects of the implementation and its efficiency will be dealt with in Sec. III. Applications to lattice models of the θ -point collapse will be presented in Sec. IV. In Sec. V, a stochastic variant of our algorithm is given that is suitable for off-lattice models. Finally, Sec. VI contains a discussion and an outlook.

II. ALGORITHM

The algorithm we propose is based on two ideas going back to the very first days of Monte Carlo simulations, the Rosenbluth-Rosenbluth (RR) method [7] and enrichment [8]. Both are modifications of SS. In SS, a chain is built by adding one new monomer after the other, placing it at a random neighbor site of the last placed monomer. In order to obtain correct statistics, an already occupied neighbor should not be *avoided*, but any attempt to place a monomer at such a place is punished by discarding the entire chain. This leads to an exponential "attrition," which makes the method useless for long chains.

In the RR method, occupied neighbors are avoided nevertheless without discarding the chain, but the bias induced thereby is counterbalanced by giving different weights to the chains produced. More precisely, assume that we work on a lattice, and that there are m_n allowed positions for the n th monomer. Compared to SS, an allowed configuration will be chosen with a relative frequency $\propto (\prod_n m_n)^{-1}$. Thus, in order to obtain an unbiased sample, each chosen configuration should be weighted with a factor

$$W_N \propto \prod_{n=1}^N m_n. \quad (1)$$

Typical chains are more dense than typical SAW's, and to compensate this, open chains have, in general, a larger weight. This renders the RR method useless for the simulation of very long ordinary (athermal) SAW's [9], but not for θ polymers. In the latter, the above weights are largely canceled by Boltzmann factors. If we choose each free neighbor k ($=1, \dots, m_n$) at the n th step with the same probability $p_k = 1/m_n$, Eq. (1) has to be replaced by

$$W_N \propto \prod_{n=1}^N w_n^{(k_n)}, \quad (2)$$

with

$$w_n^{(k)} = m_n e^{-E_k/k_B T}. \quad (3)$$

As an alternative we can include the Boltzmann factor in the probability for choosing a site, $p_k = e^{-E_k/k_B T} / \sum_{k'=1}^{m_n} e^{-E_{k'}/k_B T}$, in which case $w_n^{(k)}$ becomes independent of k ("importance sampling"),

$$w_n^{(k)} = \sum_{k'=1}^{m_n} e^{-E_{k'}/k_B T}. \quad (4)$$

The \propto signs in Eqs. (1) and (2) are to remind us that we can choose freely the constants in front of the right-hand sides, reflecting the freedom in choosing a statistical ensemble. If these constants are just N th powers, $W_N \propto z^N$, then the expectation value of $\sum_N W_N$ is directly proportional to the grand canonical partition sum with fugacity z . In the following we shall call Eq. (3) "method A," and Eq. (4) "method B." Indeed, we have an even larger choice for splitting the total weight between the probabilities p_k and the weights $w_n^{(k)}$ [10]. All we need is that the total weight of the k th choice at step n is equal to

$$p_k w_n^{(k)} = c_n e^{-E_k/k_B T}, \quad (5)$$

with a factor c_n independent of k . If c_n is also independent of n , then the sample expectation value gives directly the grand canonical ensemble. Otherwise, one has to do a (trivial) reweighting. More sophisticated choices of p_k are planned to be discussed in a forthcoming paper [11].

Precisely at the θ point the bulk of the chosen configurations more or less exactly agree with the typical configurations. Thus both versions of the RR method are efficient for θ polymers, provided the chains are not too long [1]. The critical length depends on the lattice, and for the simple-cubic lattice it is $N_c \approx 1000$ for method A [12]. For $N > N_c$ the *average* configurations are still all right, but the distribution of W_N is so wide that (rare) chains with large W_N dominate the sample nevertheless. More precisely, assume that we want to measure an observable A whose value is fluctuating

but uncorrelated with W_N . In each single event, the variance of A is σ_A^2 , and successive events are also uncorrelated. It is easily shown that in this case the variance of a sample mean $\bar{A} = \sum_{i=1}^M W_{N,i} A_i / \sum_{i=1}^M W_{N,i}$ is given by

$$\sigma_{\bar{A}}^2 = \frac{\langle W_N^2 \rangle}{M \langle W_N \rangle^2} \sigma_A^2. \quad (6)$$

Thus the error of the sample mean increases with the width of the W_N distribution. In the extreme case, the sample average is dominated by a single event, and its error is independent of the sample size.

In addition, we would like to avoid the simulation of chains with *very low* weights, since they just cost CPU time. Obviously, we need a trick to reduce the variance of the W_N distribution, if we want to use the RR method for longer chains.

The main idea in enrichment looks completely different at first sight. Here, we start from SS and counterbalance attrition by adding copies of allowed configurations at a fixed rate. Assume that we know already that the attrition would make the sample size shrink as $C_n \sim e^{-n\alpha}$. If $\alpha \approx k^{-1} \ln 2$, we would get a sample of fixed (n -independent) size if we double all surviving chains after every k th iteration. This can be done breadth first or depth first [13]. In the former case, all chains of the entire sample are simulated up to a common length n , then the entire sample is doubled, and the next k iterations are performed. In a depth-first implementation, at each time only a single configuration is dealt with, and the handling of the copies is done by recursion. This is much faster since there are much less memory accesses, and it requires much less memory, but one has to know the attrition in advance. In a breadth-first implementation, on the other hand, the attrition can be learned "on the fly," by readjusting the sample to a prefixed size at each enrichment step. If attrition is not known *a priori*, one can still use some learning strategy in a depth-first algorithm [14].

As described above, enrichment is used so that each chain contributes with the same weight, or with fixed prescribed weights. This is natural for athermal chains, and it allows perfect importance sampling in thermal systems. The different approach of the present algorithm is to use enrichment, implemented recursively, to avoid large weights in the RR method. Thus we implement the RR method by means of a recursion, and whenever the partial weight $W_n = \prod_{n'=1}^n w_{n'}^{(k_{n'})}$ exceeds some upper threshold value $W_n^>$ we double the configuration by calling the subroutine twice. At the same time, the weight of each copy is taken as half the original weight. To eliminate configurations with too small weights, we define similarly a lower threshold $W_n^<$ and *prune* the sample by eliminating every second configuration with $W_n < W_n^<$, doubling the weights of the other half. We call this the pruned-enriched Rosenbluth method (PERM). Notice that attrition can be included in the formal definition of pruning if we use method B (we just have to set $E = \infty$ for forbidden sites), and need not be treated separately.

The limits $W_n^>$ and $W_n^<$ can be chosen in advance, but in general (except very close to the θ point) it is better to adapt

them at each call of the subroutine. Notice that $W_n^>$ and $W_n^<$ can be changed arbitrarily at any time without impeding the correctness of the algorithm. But the effectivity will depend on them. Optimally they should be fixed such that the sample size will be independent of n , as this gives in general the smallest statistical errors. A good strategy is the following: we first put $W_n^<=0$ and $W_n^>$ very large number (say 10^{100}). This prevents pruning and enrichment; i.e., it corresponds to the original Rosenbluth method. After the first chain of full length has been obtained (which will not take very long), we switch to $W_n^>=c^>Z_n$ and $W_n^<=c^<Z_n$, where Z_n is the current estimate of the partition sum, and $c^>$ and $c^<$ are constants of order unity. Their optimal values are easily found by small test runs. We found that $c^>/c^<\approx 10$ gave in general good results. A pseudocode showing the main recursive subroutine of this algorithm is given in the Appendix.

With PERM we do not have *exact* importance sampling (which would mean that W_n is a unique function of n). Instead, small fluctuations of the weights are tolerated since they allow the enrichment and pruning events to be kept at a small rate. To understand this crucial point, notice that our algorithm essentially corresponds to a random walk in chain length with reflecting boundaries at $n=0$ and at $n=N$. For an algorithm to be efficient, we have to go fast towards large N and fast back, so that we create many independent configurations. Thus we need a large diffusion coefficient D , and we do not want any bias in this random walk. If we could do without any pruning, any started chain would be finished, i.e., any walk would continue until $n=N$ is reached, and D would be infinite. On the other hand, since we do not want any bias, the frequency of pruning has to match exactly that of enrichment events. Thus both pruning and enrichment should be as small as possible for efficiency. Once we have decided that we want to start a particular chain, we should not waver and should finish it.

Thus the efficiency of the method depends on a compromise between two antagonistic aspects: the method is the more efficient the smaller is the variance $\sigma_{W_n}^2$ of W_n , and the larger is the diffusion coefficient D . For θ polymers on the simple-cubic (sc) lattice we found $D>1000$ with $\sigma_{W_n}/\langle W_n \rangle < 1$. This means that our method is roughly 1000 times faster than other recursive methods with $D\approx 1$, such as incomplete enumeration [15,14] or the Beretti-Sokal (BS) [16] algorithm. The algorithm used in [5] corresponds to the largest value of D (≈ 15 on the sc lattice) possible with $\sigma_{W_n}=0$.

III. IMPLEMENTATION AND PERFORMANCE OF THE ALGORITHM

We used the above method to simulate chains near the θ point on the simple-cubic lattice, and two versions of the bond fluctuation model [17,18].

On the sc lattice, we assume an attractive energy $-\epsilon$ for each occupied nearest neighbor pair. For a monomer placed near m neighbors, we have thus a Boltzmann factor

TABLE I. Potentials between bonded (V_b) and unbonded (V) monomer pairs used in Ref. [18].

\mathbf{r}	$V_b(\mathbf{r})$	$V(\mathbf{r})$
(2,0,0)	+4.461	-3.177
(2,1,0)	+2.021	-2.275
(2,1,1)	-0.504	-1.594
(2,2,0)		-0.771
(3,0,0)	-3.275	-0.319
(2,2,1)	-3.478	-0.521
(3,1,0)	-1.707	0

$$e^{-E/k_B T} = e^{m\epsilon/k_B T} = q^m, \quad q \equiv e^{\epsilon/k_B T}. \quad (7)$$

Whenever we quote θ temperatures, we will assume units in which $\epsilon/k_B=1$.

In the bond fluctuation model, monomers are also on the sites of a simple-cubic lattice. The hard core repulsion now excludes any pairs with distance <2 lattice units. Any bond along the chain can be one of the 108 vectors formed by rotations and reflections from the set (2,0,0), (2,1,0), (2,1,1), (3,0,0), (3,1,0). This set is mainly motivated by being suitable for dynamic Monte Carlo algorithms [19]. For the present algorithm it is not particularly suitable. We chose to simulate it because of two recent papers [17,18] with which we wanted to compare our method.

In [17] the same attractive energy $-\epsilon$ was given to each pair with distance $4\leq r^2\leq 6$. Pairs with larger distances do not contribute to the energy, as do also all bonded pairs.

In [18], in contrast, the potential was much more complicated. For each bonded pair of monomers there is an energy $V_b(\mathbf{r})$. For each nonbonded pair with distance $2\leq r\leq 9$ there is a different energy $V(\mathbf{r})$. These energies are given in Table I. Notice that bonded pairs do not get a contribution from $V(\mathbf{r})$, but exclusively from $V_b(\mathbf{r})$ (this is not very clearly stated in [18]; I am indebted to the authors of [18] for sending me the values quoted in Table I). These energies are such that nonbonded interactions are strongest for the nearest pairs, while bonded interactions prefer large and intermediate distances. This will make short chains (where most interactions are bonded) more extended than long ones where nonbonded interactions dominate. For a motivation of the potential we refer to [18].

In both versions, the variance of $E/k_B T$ (where E is the monomer energy in the field of previously placed monomers) is much larger than in the sc lattice. This might be one reason why the algorithm is less effective for them. For very low energies, the variance is so big that enriching with a factor 2 is not always enough for keeping W_n below $W_n^>$. In this case, we found it necessary to make many more copies—with their weight correspondingly reduced—if $W_n \gg W_n^>$.

For verifying the self-avoiding constraint and for computing energies we have to store the chain configurations such that we have immediate access to neighboring sites. For the bond fluctuation model we simulated only chains with $N\leq 900$, since our method there is not extremely efficient (we should say, however, that the largest previous simula-

tions there had $N \leq 260$). For such chain lengths it is still feasible to store the states of all sites in a large cubic array.

For longer chains this is no longer feasible, if the chains are to be free. Thus, if we want to simulate free chains, we have to use hashing. In runs with $N = 10\,000$ on the sc lattice we used a simple hashing method where two of the three coordinates are used as keys, and linked lists are used to resolve collisions [5].

As an alternative we can keep a direct image of the chain, but we then cannot consider the chains as free any more. Instead we used periodic boundary conditions, and simulated in this way very long chains that fill the available volume to a nonzero mean density (“dense limit”). We used cubes of size L^3 with L up to 256, and chain lengths up to 10^6 .

All simulations were done on 64-bit workstations (DEC ALPHA), and we used a congruential random number generator with multiplier 13^{13} as also used in the NAG library.

In most simulations the recursions were implemented by recursive function calls in FORTRAN77. In contrast to Ref. [20], this is possible (on all work stations I know of) and efficient, and it keeps the code readable and simple. Only for the very longest chains we hand coded the recursion. This gives modest speed improvements (typically 15–25%), and it saves much memory. The latter was indeed the main reason for doing it. With recursive function calls we could only go to $N \approx 2 \times 10^5$ on our machines, which had up to 32 MB of stack. The total CPU time was roughly 3500 h.

For free chains we measured the end-to-end radius R_n , the radius of gyration $R_n^{(g)}$, and the partition sum Z_n . On the sc lattice we measured also the mean energy, and the variance of the energy distribution (which is proportional to the specific heat). For chains in the dense limit, we only measured Z_n . In addition to these physically relevant observables, we also monitored some observables that were of interest for the efficiency of the method, such as the diffusion coefficient D and the number of independent chains of lengths $n = 1, \dots, N$. Due to the very long chains involved, making histograms and reweighting them would have been impractical. Thus we either used a different run for each different temperature (for free chains), or we computed “on the fly” thermal averages for several (typically 7) different temperatures. The latter was done for the dense limit.

As a first result we verify the diffusional characteristics in n for method A on the sc lattice, precisely at $T = T_\theta (= 3.717)$. For this we used chains of length $N = 300\,000$ on a lattice with $L = 512$ (this was affordable since we did not need very high statistics for this). We measured the average squared length difference $\langle (n_2 - n_1)^2 \rangle$ after t function calls. To avoid end effects, the averaging was done only over chains with $n_2 \in [50\,000, 250\,000]$. We expect

$$\langle (n_2 - n_1)^2 \rangle \approx Dt \quad (8)$$

for large t . For small values of t (more precisely, for $t < D$) we might expect deviations since the evolution is not diffusive. Instead, the length *increase* is essentially linear in time, while the *decrease* happens in jumps of size $\approx D$. But no such deviations were observed. In Fig. 1 we see a nearly

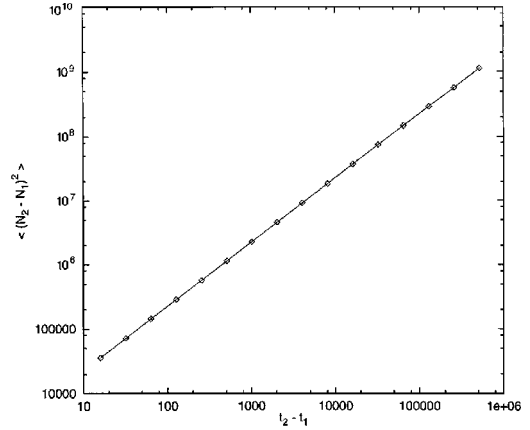


FIG. 1. Log-log plot of the mean square length difference $\langle (n_2 - n_1)^2 \rangle$ vs the number $t = t_2 - t_1$ of function calls in between. This figure is for method A on the sc lattice, but similar plots (with smaller slopes) were found in all other cases. The upper and lower thresholds for enrichment and pruning were 2.3 and 0.44, respectively.

perfect straight line with unit slope on a log-log plot, indicating that Eq. (8) holds in both regimes. The value of D is 2230 ± 100 .

The ratio $W_n^>/W_n^<$ used in Fig. 1 was 5.23. The measured value $\langle W_N^2 \rangle / \langle W_N \rangle^2$ for this ratio was 1.1158, whence the lack of perfect weight balancing has only a very minor effect, increasing the error by $\approx 6\%$. Indeed we found that D was nearly constant over the range of $4 < W_n^>/W_n^< < 20$, while $\langle W_N^2 \rangle / \langle W_N \rangle^2$ was slowly increasing with this ratio.

Actually, using the ratio $\langle W_N^2 \rangle / \langle W_N \rangle^2$ as an indicator for the efficiency of the method is somewhat misleading since the chains are not independent. Rather, all chains generated during one “tour” (i.e., between two successive returns to the main program) are correlated. To obtain a correct upper

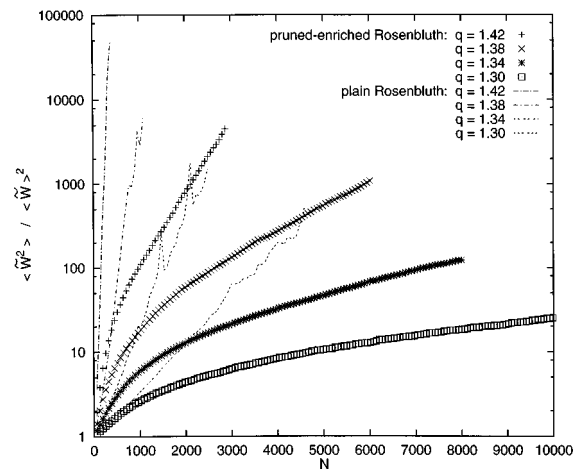


FIG. 2. Normalized second moments of tour weights vs chain length N . For the RR method, a tour is just a single chain. For PERM, it is the set of all chains generated between two successive returns to the main program in a recursive implementation. In PERM, the thresholds were set to $W^< = 0.3$, $W^> = 3$, and fugacities were chosen such that the number of generated chains was roughly independent of N .

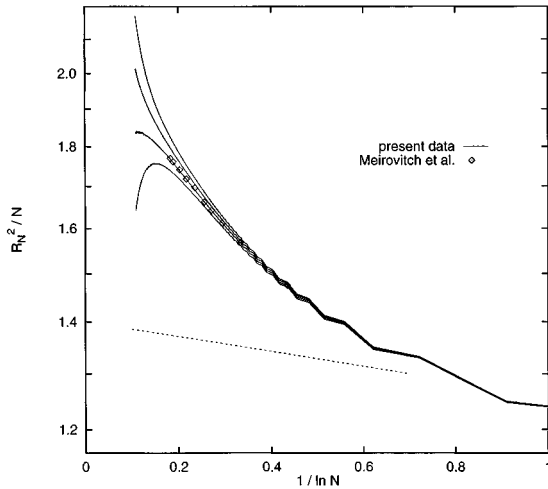


FIG. 3. Logarithms of the swelling factor R_N^2/N for the end-to-end distance, plotted against $1/\ln N$. The four continuous curves are for Boltzmann factors $q = 1.315, 1.310, 1.305,$ and 1.300 (bottom to top). The slope of the dashed line is the prediction from field theory. The diamonds are from [22].

bound on the error of any observable by means of Eq. (6), we should thus use weights of entire tours instead of weights of single chains. Let us denote the weight by which a tour contributes at length n by \tilde{W}_n . In Fig. 2 we show the ratios $\langle \tilde{W}_n^2 \rangle / \langle \tilde{W}_n \rangle^2$ for several temperatures, and compare them to the analogous ratios for the RR method without pruning and enrichment. We see that the ratios are indeed much smaller for PERM than for the RR algorithm, demonstrating thereby its greater efficiency. We also see that they are smallest near the θ point, and increase strongly at low temperatures. This points at the main limitation of PERM: like all other known algorithms it becomes inefficient for long chains and at low temperatures, since the generated chains are strongly correlated and state space is no longer evenly sampled.

We performed similar (but less complete) analyses also for method *B*, and for the bond fluctuation model. In all these cases we found D to be much smaller (≤ 100 , typically), but in all cases the behavior was diffusive.

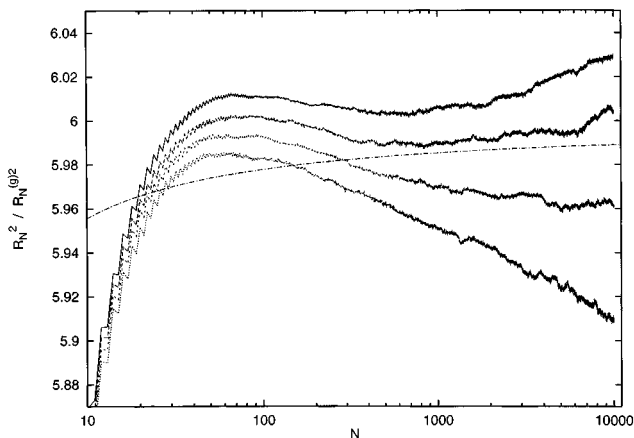


FIG. 4. Ratios $R_N^2/R_N^{(g)2}$ against $\ln N$. The values of q are the same as in Fig. 3, with large (small) q at the bottom (top). The dashed-dotted line is the field theory prediction.

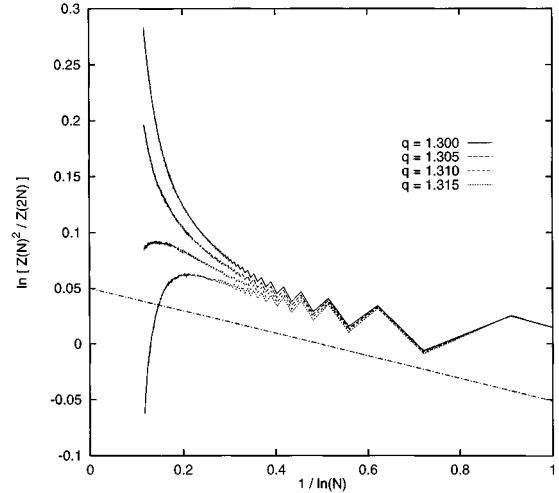


FIG. 5. Logarithms of Z_N^2/Z_{2N} against $1/\ln N$. The slope of the dashed-dotted line is again the field theory prediction.

Surprisingly, we found that in some cases method *A* performed better than method *B* (sc lattice), while in other cases (bond fluctuation model) the opposite was true. We have no good explanation for this fact. Obviously, the impressive performance for the sc lattice depends on the fact that θ chains on this lattice are nearly ideal random walks, and the deviations from the ideal case are taken care of by small deviations from uniform weights that do not require much pruning or splitting. This is not so for either version of the bond fluctuation model, where some Boltzmann factors are very large at the θ point. For the model of [18], the best performance was indeed obtained for an intermediate method where part of the Boltzmann factor was treated as in method *A*, and the rest as in method *B*.

IV. RESULTS

A. Simple-cubic lattice, free chains

In a first set of runs we simulated free chains at Boltzmann factors $q \equiv e^{1/k_B T} = 1.300, 1.305, 1.310,$ and 1.315 . These are the same as studied also in [5]. We performed all measure-

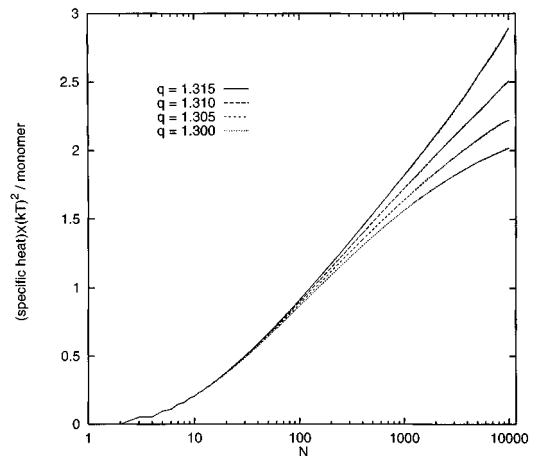


FIG. 6. Variance of the energy divided by the chain length N , plotted against $\ln N$.

ments also made already in [5], but now with $N=10\,000$ instead of 5 000, and with higher statistics. More precisely, for each $n<5000$ the number of generated walks is only roughly 1/5 of the number of walks generated in [5], but due to the much larger value of D the number of *independent* walks of length 5000 is roughly 50 times larger. Accordingly, the present error bars at $N=5000$ are roughly 3 times smaller than those in [5]. They were obtained by dividing the tours into independent bins, and estimating the bin-to-bin fluctuations. At $N=10\,000$ (were errors are largest) the relative errors are 0.0014 for R_N^2 , 0.001 for $R_N^{(g)2}$, and 0.003 for Z_N . The results shown in Figs. 3–6 are based on both the present data and those of [5].

Results for the average squared end-to-end distance R_N^2 are shown in Fig. 3. More precisely, we plot there R_N^2/N versus $1/\ln N$, since the field theoretic renormalization group predicts a behavior [6]:

$$R_N^2/N = \text{const} \times \left(1 - \frac{37}{363 \ln N} \right). \quad (9)$$

As already found in [5], our data do not agree with this prediction. Thus there are strong corrections to the leading asymptotic behavior, most likely due to weak effective three-body forces, and strong effective n -body forces for $n>3$ [21].

The ratio $R_N^2/R_N^{(g)2}$, where $R_N^{(g)}$ is the gyration radius, is shown in Fig. 4. Again the theoretical prediction [6] is shown as a dashed-dotted line, and again we find no good agreement for $N<500$ —as was found already in [5]. But the present data do agree with [6] for $N>1000$ within the error bars. In particular—and in spite of the nonmonotonic behavior for smaller N — $R_N^2/R_N^{(g)2}$ seems to approach its asymptotic value 6 (for $T=T_\theta$) from *below*, in agreement with the prediction.

In Fig. 5 we show the ratio Z_N^2/Z_{2N} . We have chosen this as it is independent of the chemical potential, and thus a prediction similar to Eq. (9) exists for it from field theory [6,5]. As already found in [5], the disagreement with the prediction is less severe than for R_N^2/N , but now we definitely find that there is a disagreement, and nonleading terms are important.

Finally, the specific heat (or rather the variance of the energy) per monomer is plotted in Fig. 6. Again we confirm the trend seen already at smaller chain lengths [22,5]: the specific heat increases much faster with N than predicted by field theory. The latter gives $c \sim (\ln N)^{3/11}$, while the simulations show rather $c \sim \ln N$. But we now see that the latter is not quite correct, and c increases indeed slower than linearly with $\ln N$. Thus it seems not unlikely that finally the field theory prediction is reached, but only at *very* long chains.

The results shown in Figs. 3–6 support fully the previous estimate $q_\theta \approx 1.308$. In spite of the longer chains and the higher statistics, they do not give a much more precise estimate. The reason is the same as that which also prevented more definite conclusions in [5]: we cannot rely on field theory for the extrapolation to $N=\infty$, and this extrapolation is nontrivial.

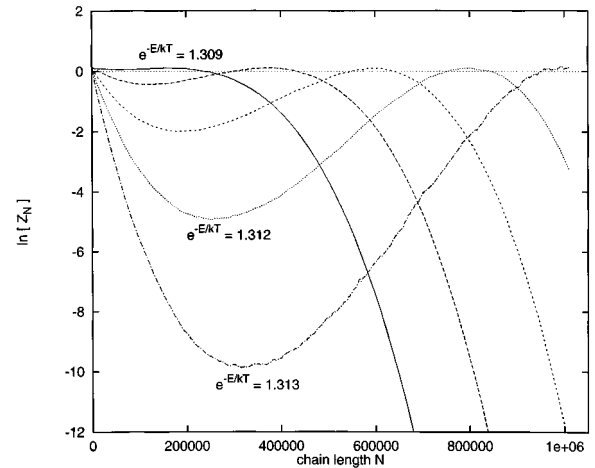


FIG. 7. Logarithms of the partition function versus chain length at 5 equally spaced values of $q = e^{\epsilon/k_B T}$, 1.309, 1.310, . . . , 1.313. The lattice had 256^3 sites with helical boundary conditions.

B. Simple-cubic lattice, dense limit

In [5] we had found even more embarrassing results for $T < T_\theta$. From simulations of free chains it was virtually impossible to exclude the possibility that the θ point is first order in the sense that the monomer density in a large globule jumps discontinuously to a finite nonzero value when passing T_θ . Notice that there exist indeed models in which the θ -point collapse is first order [21], but it is generally believed that this is not true for real polymers.

To decide this question, we had simulated in [5] very long chains in finite volumes (“dense limit”). From these we had concluded that the transition is indeed smooth, but the monomer density in the interior of a blob is not governed by the mean field exponent:

$$\rho(T) \sim (T_\theta - T)^{0.7} \quad (10)$$

instead of $\rho \sim (T_\theta - T)^1$.

In order to investigate this problem further we made simulations with the present improved algorithm. In contrast to the simulations in [5] we now used n -dependent thresholds $W_n^>$ and $W_n^<$ so that the samples had roughly n -independent sizes. In this way it was possible to generate tours with extremely long chains, which were nevertheless finished within reasonable CPU times. We thereby avoid completely the bias discussed in detail in [5,23]. We used helical boundary conditions, which should give results indistinguishable from periodic ones.

In Fig. 7 we show the free energies (or, more precisely, the logarithms of the partition sums) for 5 different temperatures as functions of N . The common lattice size for all 5 curves is $L^3 = 256^3$. All temperatures are slightly less than T_θ , and fugacities were chosen such that the two maxima at $N=O(1)$ and at $N=O(L^3)$ had the same height. Assuming that plots of this type scale with L , we can take the positions of the latter maxima to estimate the densities in a pressure free medium in the limit $N \rightarrow \infty$, i.e., in the central region of an infinitely large free globule. From Fig. 7 and similar plots

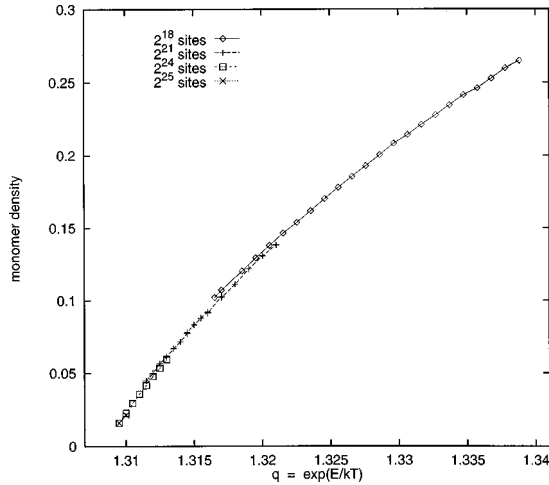


FIG. 8. Monomer densities at the transition from $\rho=0$ to $\rho>0$ on finite but large lattices, plotted against q .

for different values of L we obtain in this way the estimates shown in Fig. 8. In this figure we see slight violations of scaling since the estimates do depend slightly on L . But these scaling violations seem to decrease with L . We see unambiguously that $\rho \equiv N/L^3 \rightarrow 0$ for $T \rightarrow T_\theta$. Indeed, these simulations give our most precise estimate for the collapse temperature:

$$q_\theta = 1.3087 \pm 0.0003, \quad T_\theta = 3.717 \pm 0.003. \quad (11)$$

But it is still not easy to decide whether or not ρ increases linearly with $(T_\theta - T)$. A least square fit to the data of Fig. 8 would be compatible with the power law (10). But a closer inspection shows that there are systematic deviations from such a fit, indicating that the effective exponent increases to 1 as $T \rightarrow T_\theta$. Thus we suggest that Eq. (10) does *not* represent the true asymptotic behavior, and mean field theory does become correct for $T \rightarrow T_\theta$ (although the numerical evidence for this is rather shaky).

The same conclusion is reached when studying the free energy $F = -\ln Z_N$ at fixed temperature. As discussed in [5], mean field theory assumes that F is a cubic function of ρ , with the linear term vanishing at the critical fugacity and the quadratic vanishing in addition at T_θ . Thus at the critical fugacity and at $T = T_\theta$ it is predicted that $F \sim L^3 \rho^3$. In Fig. 9 we show $-\ln Z_N$ against N/L^2 for $q = 1.3085$. Within the statistical error this is the θ point, and the above argument suggests that the curves should collapse to a cubic $\ln Z_N = \text{const} \times (N/L^2)^3$ if mean field theory is exact. We see that this is not quite correct, but the discrepancies diminish with increasing L . Even more importantly, while each of the curves can be fitted by a power law, the powers decrease from 3.65 for $L = 64$ to 3.22 for $L = 256$. This suggests that the exponent does indeed converge to 3 for $L \rightarrow \infty$, and mean field theory does predict the correct power laws albeit with very large finite size corrections.

C. The Bond fluctuation model

We have studied this model since there have been suggestions in the literature [18] that the very large logarithmic

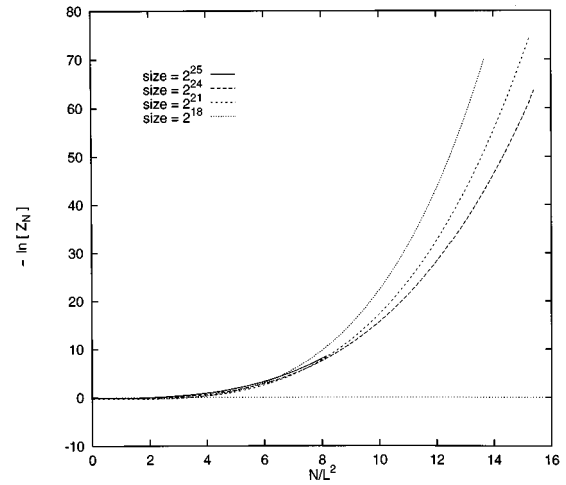


FIG. 9. $F = -\ln Z_N$ against N/L^2 for $q = 1.3085$, and for the same 4 lattice sizes as in Fig. 8. According to mean field theory, the data should collapse onto a cubic curve.

corrections found in [5] are an artifact of the simple-cubic lattice, and would not show up in off-lattice models or in lattice models with very high coordination number.

In Figs. 10 and 11 we show the ratios $R_N^{(g)2}/N$ and $R_N^2/R_N^{(g)2}$, respectively, against $1/\ln N$, for the model of Wittkopp *et al.* [18]. The estimated θ temperature is given by these authors as $\epsilon/k_B T_\theta = 0.214 \pm 0.008$, i.e., $T_\theta = 4.67 \pm 0.17$ for $\epsilon/k_B = 1$. This was obtained from simulations up to $N = 100$. Comparing this with Figs. 10 and 11 where we used chains with $N \leq 900$, we see that this estimate would be correct if the behavior of $R_N^{(g)2}/N$ would not change for $N > 100$, but this is not the case. Thus, T_θ is definitely larger than 4.7, and $R_N^{(g)2}/N$ increases considerably at the true θ point. Our best estimate of the latter is $T_\theta = 5.04 \pm 0.03$, about 2 standard deviations higher than the estimate of [18].

In view of the claim of [18] that conventional mean field scaling holds good in their model (see Fig. 5 in the second paper of [18]), we also simulated at lower temperatures, and plotted $(T_\theta - T)(R_N^{(g)2}/N)^3 \sqrt{N}$ against $(T_\theta - T)\sqrt{N}$; see Fig. 12. We found indeed much better data collapse than in the sc

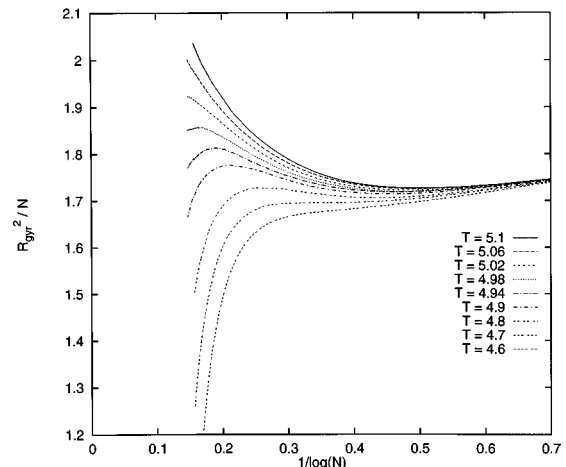


FIG. 10. Gyration radius swelling factors $R_N^{(g)2}/N$ against $1/\ln N$ for the model of [18].

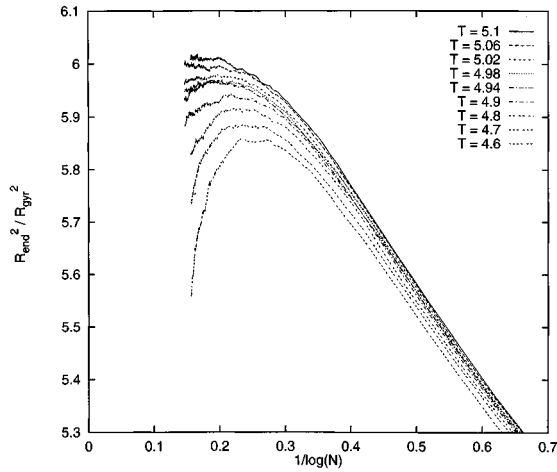


FIG. 11. Ratios $R_N^2/R_N^{(g)2}$ against $1/\ln N$. The model and the values of T are the same as in Fig. 10.

lattice model, but this is easily explained as a finite N effect. The potential in [18] was chosen such that the average bond length increases at low temperatures. This suppresses the increase of $R_N^{(g)2}$ with T for short chains and for very low temperatures, leading thereby to improved scaling. But this should have a smaller effect for long chains close to T_θ , and we see indeed indications in Fig. 12 that the scaling violations increase as we approach the scaling limit (where they should *decrease*, of course, if mean field scaling would hold).

A similar analysis was made for the version of the bond fluctuation model used in [17]. Ratios $R_N^{(g)2}/N$ and $R_N^2/R_N^{(g)2}$ against $1/\ln N$ (with N up to 600) are now plotted in Figs. 13 and 14. These plots again look very much like their counterparts for the model of [18] and for the sc lattice. We do not show the scaling plot analogous to Fig. 12 since we did not perform systematic simulations far below T_θ . But already the data shown in Fig. 13, when plotted as in Fig. 12, show much larger scaling violations, similar to those found

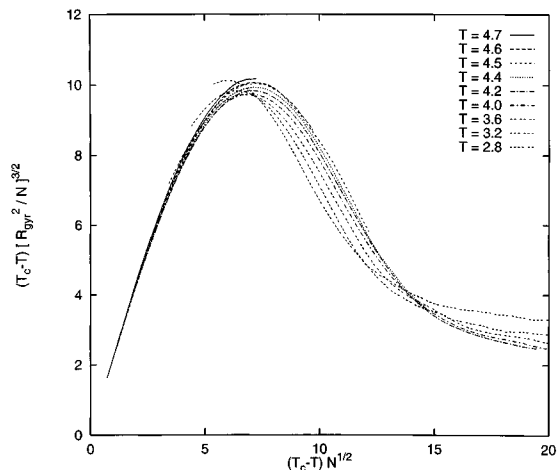


FIG. 12. Scaling plot of $(T_\theta - T)(R_N^{(g)2}/N)^{3/2}$ against $(T_\theta - T)\sqrt{N}$ for the model of [18]. If mean field theory were exact, the data would collapse onto a single curve. We see very small violations indeed, which seem to increase, however, as we approach the θ point.

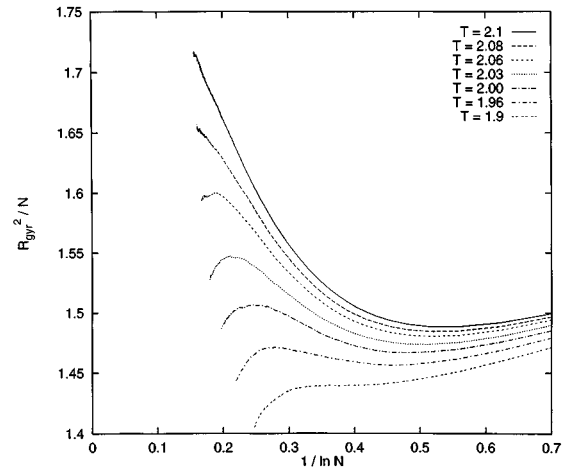


FIG. 13. Analogous to Fig. 10, but for the interactions of [17].

in [5]. This was to be expected according to the discussion in the previous paragraph, since the potential between bonded monomers chosen in [17] does not depend on the bond length. Our estimate for the collapse temperature from Figs. 13 and 14 is $T_\theta = 2.10 \pm 0.01$, as opposed to the value 2.02 ± 0.02 obtained in [17] from chains of length ≤ 150 . Again we see the strong effect of corrections to mean field scaling.

V. STOCHASTIC PERM: APPLICATIONS TO OFF-LATTICE CHAINS

As described above, PERM is applicable only to lattice chains. More precisely, for implementing the Rosenbluth-Rosenbluth trick, we have to know the weights for all possible next moves. This is in general only possible when there is a finite number of such moves. For off-lattice models, it is only feasible in very simple two-dimensional models [24], and even then it is not clear that the effort is worthwhile.

But it is rather straightforward to put forward a stochastic version of the Rosenbluth-Rosenbluth trick, where these weights are estimated from finite samples. In order to add a monomer, we thus first choose randomly s sites. For maximal efficiency, these sites should be distributed according to

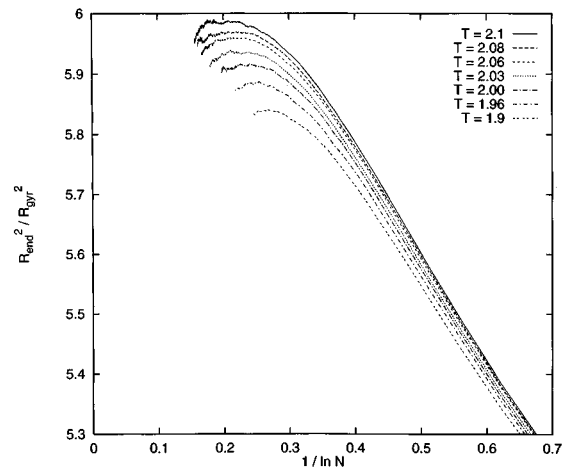


FIG. 14. Analogous to Fig. 11, but for the interactions of [17].

a measure (“guiding field”) that is not too different from the estimated true Boltzmann weight [25–27]. For each of them we estimate an acceptance factor, either including the Boltzmann factor (method *B*) or not (method *A*). The actual site of the next monomer is then chosen among these sites, and the actual weight is computed so that the total sample is unbiased. Finally, the enrichment and pruning steps are done exactly as before.

We have implemented this for a model studied extensively by Freire and others [28–32]. In this model, non-bonded monomers interact with a Lennard-Jones potential with strength ϵ and range σ , $V(r) = 4\epsilon[(\sigma/r)^{12} - (\sigma/r)^6]$. Bonded monomers do not feel any mutual hard or soft core repulsion. Instead, they interact exclusively through a Gaussian potential whose strength is proportional to T , so that its associated Boltzmann factor is independent of T , and whose range is such that the average squared bond length is 1. Following Refs. [28–32], the Lennard-Jones (LJ) range was kept fixed at $\sigma = 0.8$, and it was not truncated at large distances.

We do not consider this model as particularly realistic (in view of the absence of any repulsion between bonded neighbors) nor convenient (due to the absence of truncation of the LJ potential). We chose it solely because of claims in [31,32] that it does not show the large logarithmic corrections found in [5]. It was claimed there that logarithmic corrections found in this model are fully in agreement with a finite- N extrapolations of the asymptotic results of [6]. For instance, the equation for the gyration radius analogous to Eq. (9) is replaced by

$$R_N^{(g)2}/N = A_0 \left(1 - \frac{148\pi}{33} h(N) \right), \quad (12)$$

with

$$A_0 = 1 + \frac{16}{33} \pi z_3, \quad h(N) = \frac{z_3}{1 + 44\pi z_3 \ln(N/n_0)} \quad (13)$$

and with $z_3 = 0.458$ and $n_0 = 11.5$.

In stochastic PERM, the main parameter is the number s of trials at each monomer insertion. We have tried $s = 1, 2,$ and 3 . All gave nearly the same efficiency. Clearly, the effort increases with s , but the efficiency *per added monomer* increases also, so that the overall efficiency per unit CPU time was the same within statistical errors. The results were of course also the same for all s ; i.e., we checked that the method did not induce systematic errors.

In Figs. 15 and 16 we show again results for $R_N^{(g)2}/N$ and $R_N^2/R_N^{(g)2}$, respectively, against $1/\ln N$. Chain lengths were $N \leq 500$, and we used $s = 2$. In Fig. 15 we also plotted the modified prediction (12), and the simple asymptotic prediction obtained by putting $n_0 = 1$ and $\ln N \gg 1$. We see that our data indeed coincide roughly with Eq. (12) for $N > 100$ (this was also the range used in [31]), provided $\epsilon/k_B T_\theta = 0.240$. This is also the estimate of [31]. Thus our simulations agree at this coarse level. But Fig. 15 should not leave any doubt

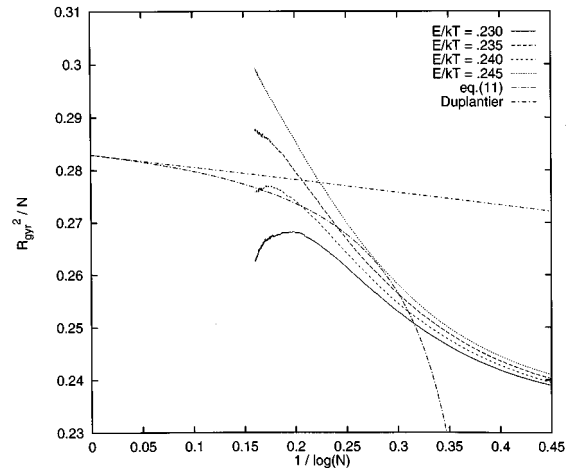


FIG. 15. Analogous to Fig. 10, but for the off-lattice model of Freire *et al.* The smooth lines are the asymptotic (“naive”) prediction from field theory [6] and its finite- N modification given in Eqs. (12) and (13).

that Eq. (12) does *not* give the asymptotic behavior. This is also confirmed by Fig. 16, although we should be careful in the interpretation of this figure: as found for the sc lattice (Fig. 4), $R_N^2/R_N^{(g)2}$ can have a maximum at some finite value of N , and approach its asymptotic value from below nevertheless. Both figures suggest that $\epsilon/k_B T_\theta = 0.232 \pm 0.002$. Accepting this, we conclude that also the analysis of the second virial coefficient in [32] is misleading (since it was based on a wrong value of T_θ), and that logarithmic corrections are indeed much larger than predicted by the field theoretic renormalization group.

VI. SUMMARY AND OUTLOOK

As far as the physics of the θ -point collapse is concerned, the results of the present paper are easy to summarize: while we fully confirm the conventional view that the θ point is a tricritical point and thus mean-field-like in three dimensions, we also confirm the very large corrections to mean field behavior found in [5]. The only point where we deviate from

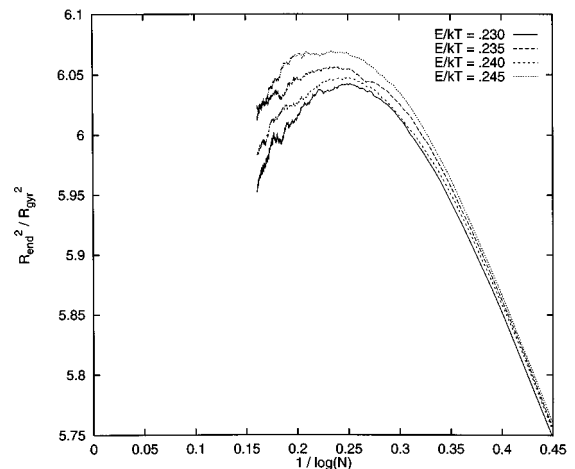


FIG. 16. Analogous to Fig. 11, but for the off-lattice model of Freire *et al.*

[5] is in the T dependence of the monomer density inside a large globule for $T < T_\theta$. Although we see also there large deviations from mean field behavior, it seems now that these corrections are nonleading, and the asymptotic behavior is $\rho \sim T_\theta - T$.

We derived this from simulations of very long chains with very high statistics on the sc lattice, but we verified that the same qualitative behavior holds also for off-lattice chains and for chains on lattices with very high coordination numbers. This rules out speculations that the large corrections found in [5] might be lattice artifacts. As a side result we found that T_θ was underestimated in *all* previous papers.

Maybe more interesting is the algorithmic aspect of the present work. Although the general structure of the algorithm is similar to other recent chain growth algorithms for polymeric systems [15,16,25,26,5], the details are quite different. At least in one case—the θ -point collapse on the sc lattice—this has boosted the efficiency enormously. In the other cases studied in the present paper the method is also more efficient than other known methods, but not by very large margins.

A non-negligible advantage of the present method over those of [15,16,5] is that the parameters steering the growth of the sample can now be adapted automatically at any time, while they had to be carefully chosen in advance for the previous algorithms. Making good choices had not always been easy. We should mention that also breadth-first implementations [25,26] have no problem with choosing these parameters. But they are much slower and/or storage demanding on single-processor machines since a large number of chains (typically $>10^4$) must be kept simultaneously in memory. Breadth-first implementations could, however, be efficient on massively parallel systems [33].

But we should point out that we have not yet pushed the method to its very limits. In particular, we have used only very simple guiding fields: we have either selected from all allowed moves with the same probability, or we have chosen them according to some Boltzmann weights. More alternatives are possible and rather straightforward to implement. For stretched polymers, i.e., we could implement a directional bias. The same could be useful for polymers in shear flow or for polyelectrolytes.

Another possibility would be to look ahead as in the scanning method [22]. There, the likely “success” of a move is estimated by trying all combinations of k future moves instead of only the next one. Although this can lead to large improvements in terms of efficiency per added monomer, it also leads to a much larger numerical effort per monomer. An alternative that gives similar efficiency without any substantial increase in effort is to look *back* instead. More precisely, during a short test run one puts up a histogram of success probabilities of next moves, conditioned on the k previous moves. We implemented this for two-dimensional athermal SAW’s on the square and honeycomb lattices. Using histograms of depth $k=10$ and 15 we obtained roughly one order of magnitude in speedup. Details of this “Markovian guide PERM” will be presented elsewhere [11].

We should point out that the applicability of PERM is not restricted to single polymer chains. The most straightforward extension is to semi-dilute solutions of long polymers. A

study of the critical behavior of such systems on the sc lattice is in progress [34]. A rather unusual aspect of our method in this context is that it does not allow us too many chains (the maximal number is ≈ 200), but it puts hardly any constraint on their lengths. This results simply from the fact that the critical temperature approaches T_θ when $N \rightarrow \infty$, whence our method becomes increasingly efficient in this limit. Test runs with 128 chains of length 512 were very encouraging.

Basically, PERM uses the fact that the space of configurations is given a rooted tree topology, and moves are performed only along the branches of the tree (“tree moves”). No direct moves between branches (“off-tree moves”) are allowed. Thus no loops are possible, and detailed balance is trivially satisfied. In principle this can always be done for any problem with discrete configuration space (for continuous space, the stochastic method of Sec. V can be used), but in general the method will not be efficient. Obviously the efficiency depends on the importance of tree moves versus off-tree moves. While the latter are very restricted and thus less important for single chains, they are more important for multichain systems and even more so for systems of independent particles.

An important aspect that explains the efficiency of PERM is the fact that it is *not Markovian* on the space of configurations. In a Markov process on a tree, the decisions whether a branch is to be pruned or not, and whether an enrichment step is inserted (i.e., whether more than one daughter node of the tree is visited) has to be independent of the previous history. For polymers, the resulting algorithm is the Beretti-Sokal algorithm [16]. As we have seen in Sec. III, on the sc lattice it is less efficient than PERM by about 3 orders of magnitude. Roughly, in an algorithm without memory about previous moves we have to start many new directions, since we make each time a random decision as to whether this new direction should be continued or not.

This aspect of PERM seems to be in contrast to *all* other Monte Carlo methods for equilibrium systems. Indeed, usually the first requirement for any Monte Carlo algorithm is that it is Markovian, since otherwise it would be hard to verify detailed balance. As we have seen, we do not have this problem in PERM. Maybe the most interesting challenge is to devise non-Markovian Monte Carlo algorithms that have the same “inertia” as PERM but that do not require a tree structure. In a loose sense, such algorithms would be similar to cluster algorithms [35] since moves would be correlated instead of being done independently. They could be extremely efficient for virtually all equilibrium problems.

ACKNOWLEDGMENTS

I am indebted to many colleagues for discussions, in particular to U. Bastolla, H. Frauenkron, R. Hegger, T. Lengauer, W. Nadler, H. Rieger, K. Schilling, and W. Zimmermann. This work was supported by the DFG within the Graduiertenkolleg “Feldtheoretische und numerische Methoden in der Elementarteilchen- und Statistischen Physik,” and within Sonderforschungsbereich 237.

APPENDIX

Here we show a pseudocode for the central subroutine:

```

subroutine STEP(x,n)
  choose x' near x with density ρ(x-x')           in the simplest case, ρ(x)=1/m_n δ_{|x|,1}
  w_n = c_n ρ(x-x')-1 exp(-E(x')/k_B T)          if c_n = const → grand canonical
  W_n = W_{n-1} w_n
  do statistics:
    Z_n = Z_n + W_n                                partition function
    R2_n = R2_n + x'2 W_n                          end-to-end distance sums
    t = t + 1                                       total number of subroutine calls
  etc.
end do
if n < N_max and W_n > 0 then
  W^> = c^> Z_n / Z_1                               adaption of W^> (optional)
  W^< = c^< Z_n / Z_1                               adaption of W^< (optional)
  if W_n > W^> then
    W_n = W_n / 2
    call STEP(x', n+1)
    call STEP(x', n+1)                             enrichment
  else if W_n < W^< then
    W_n = 2 W_n
    draw ξ uniformly ∈ [0,1]
    if ξ < 1/2 call STEP(x', n+1)                   prune with probability 1/2
  else
    call STEP(x', n+1)                             normal Rosenbluth step
  end if
end if
return
end

```

It is called from the main routine with arguments $\mathbf{x}=0$, $n=1$. To compute the energy $E(\mathbf{x}')$ of the newly placed monomer in the field of the earlier ones, one can use either bit maps (in lattice models with small lattices), hashing, or neighborhood lists. If none of these is suitable, $E(\mathbf{x}')$ has to be computed as an explicit sum over all earlier monomers. W_n , c_n , Z_n , and $R2_n$ are global arrays, while $c^>$, $c^<$, N_{\max} , and t are global scalar variables. In easy cases, the lines involving $c^>$ and $c^<$ can be dropped, and $W^>$, $W^<$ are global scalars. In more sophisticated implementations, ρ , $c^>$, and $c^<$ will depend on n and/or on the configuration of the monomers with indices $n' < n$. Good choices for these functions may be crucial for the efficiency of the algorithm, but are not important for its correctness.

The last statement should be qualified more precisely: if

the subroutine is called once from the main routine, the returned variable Z_n is a random variable whose average value is exactly equal to the partition sum; and if the algorithm is repeated sufficiently often, the sample average of any measured observable (such as $R_n^2 = R2_n / Z_n$) converges towards its true average. But the fluctuations at finite sample sizes can be extremely non-Gaussian for large systems and for bad choices of ρ , $W^>$, and $W^<$. In particular, they are skewed with a long tail of rare events with large statistical weights. In extreme cases, most of the statistical weight could be concentrated in events that are so rare that no such event is observed at all in the sample at hand. Unless sufficient care is taken, this can easily lead to serious underestimation of partition sums and of statistical errors.

[1] K. Kremer and K. Binder, Phys. Rep. **7**, 259 (1988).

[2] M. Lal, Mol. Phys. **17**, 57 (1969).

[3] N. Madras and A. D. Sokal, J. Stat. Phys. **50**, 109 (1988).

[4] P. G. de Gennes, *Scaling Concepts in Polymer Physics* (Cornell University Press, Ithaca, 1979).

[5] P. Grassberger and R. Hegger, J. Chem. Phys. **102**, 6681 (1995).

[6] B. Duplantier, J. Chem. Phys. **86**, 4233 (1987).

- [7] M. N. Rosenbluth and A. W. Rosenbluth, *J. Chem. Phys.* **23**, 356 (1955).
- [8] F. T. Wall and J. J. Erpenbeck, *J. Chem. Phys.* **30**, 634 (1959); *ibid.* **30**, 637 (1959).
- [9] J. Batoulis and K. Kremer, *J. Phys. A* **21**, 127 (1988).
- [10] P. Grassberger and R. Hegger, *J. Phys. A* **27**, 4069 (1994).
- [11] P. Grassberger (unpublished).
- [12] W. Bruns, *Macromolecules* **17**, 2826 (1984).
- [13] R. Tarjan, *SIAM J. Comput.* **1**, 146 (1972).
- [14] P. Grassberger, *J. Phys. A* **26**, 1023 (1993).
- [15] S. Redner and P. J. Reynolds, *J. Phys. A* **14**, 2679 (1981).
- [16] A. Beretti and A. D. Sokal, *J. Stat. Phys.* **40**, 483 (1985).
- [17] N. B. Wilding, M. Müller, and K. Binder, *J. Chem. Phys.* **105**, 802 (1996).
- [18] M. Wittkopp, S. Kreitmeier, and D. Goeritz, *Phys. Rev. E* **53**, 838 (1996); *J. Chem. Phys.* **104**, 3373 (1996); *Macromolecules* **29**, 4754 (1996).
- [19] I. Carmesin and K. Kremer, *Macromolecules* **21**, 2819 (1988); H. P. Deutsch and K. Binder, *J. Chem. Phys.* **94**, 2294 (1991); W. Paul, K. Binder, D. W. Heermann, and K. Kremer, *J. Phys. II (France)* **1**, 37 (1991).
- [20] W. J. Thompson, *Comput. Phys.* **10**, 25 (1996).
- [21] P. Grassberger and R. Hegger, *J. Phys. A* **29**, 279 (1996).
- [22] H. Meirovitch and H. A. Lim, *J. Chem. Phys.* **92**, 5144 (1990).
- [23] P. Grassberger and R. Hegger, *Ann. Phys. (Leipzig)* **4**, 230 (1995).
- [24] N. C. Smith and R. J. Fleming, *J. Phys. A* **8**, 929 (1975).
- [25] T. Garel and H. Orland, *J. Phys. A* **23**, L621 (1990).
- [26] P. G. Higgs and H. Orland, *J. Chem. Phys.* **95**, 4506 (1991).
- [27] P. Grassberger and R. Hegger, *J. Phys. C* **7**, 3089 (1995).
- [28] J. J. Freire, J. Pla, A. Rey, and R. Prats, *Macromolecules* **19**, 452 (1986).
- [29] J. J. Freire, A. Rey, M. Bishop, and J. H. R. Clarke, *Macromolecules* **24**, 6494 (1991).
- [30] A. M. Rubio, J. J. Freire, J. H. R. Clarke, C. W. Yong, and M. Bishop, *J. Chem. Phys.* **102**, 2277 (1995).
- [31] C. W. Yong, J. H. R. Clarke, J. J. Freire, and M. Bishop, *J. Chem. Phys.* **105**, 9666 (1996).
- [32] A. M. Rubio and J. J. Freire (unpublished).
- [33] I am indebted to T. Lengauer for this remark.
- [34] H. Frauenkron and P. Grassberger (unpublished).
- [35] R. H. Swendsen and J-S. Wang, *Phys. Rev. Lett.* **58**, 86 (1987).