of independent bonds. Also the corresponding eigenvectors and eigenrows may be evaluated by suitable combination of those given by eqs 20–23, in accordance with the formulation recently given.[11]

References and Notes

(1) Pajot-Augy, E.; Bokobza, L.; Monnerie, L.; Castellan, A.; Bouas-Laurent, H.; Millet, C. *Polymer* **1983,** *24,* 117. Bokobza, L.; Pajot-Augy, E.; Monnerie, L.; Castellan, A.; Bouas-Laurent, H. *Polym. Photochem.* **1984,** *5,* 191. Pajot-Augy, E.; Bokobza, L.; Monnerie, L.; Castellan, A.; Bouas-Laurent, H. *Macromolecules* **1984,** *17,* 1490. Pham-Van-Cang, C.; Bokobza, L.; Monnerie, L.; Vandendriessche, J.; De Schryver, F. C. *Polym. Commun.* **1986,** *27,* 89. Bokobza, L.; Monnerie, L. In *Photophysical and Photochemical Tools in Polymer Science*; Winnik, M. A., Ed.; D. Reidel: Dordrecht, The Netherlands, 1986. Bokobza, L.; Pham-Van-Cang, C.; Monnerie, L.; Vandendriessche, J.; De Schryver, F. C. *Polymer* **1989,** *30,* 45.

(2) Vandendriessche, J.; Van der Auweraer, M.; De Schryver, F. C. *Bull. Soc. Chim. Belg.* **1985,** *94,* 991.

(3) Glarum, S. H. *J. Chem. Phys.* **1960,** *33,* 639.

(4) Anderson, J. E.; Ullman, R. *J. Chem. Phys.* **1967,** *47,* 2187.

(5) Anderson, J. E. *J. Chem. Phys.* **1967,** *47,* 4879.

(6) Sillescu, H. *J. Chem. Phys.* **1971,** *54,* 5.

(7) Bahar, I.; Erman, B. *Macromolecules* **1987,** *20,* 1368. Bahar, I.; Erman, B.; Monnerie, L. *Macromolecules* **1989,** *22,* 2403.

(8) Kramers, H. A. *Physica* **1940,** *7,* 284.

(9) Doolittle, A. *J. Appl. Phys.* **1951,** *22,* 1471.

(10) Schwarz, G. *Rev. Mod.* Phys. **1986,** *40,* 206.

(11) Kloczkowski, A.; Mark, J. E.; Bahar, I.; Erman, B. *J. Chem. Phys.* **1990,** *92,* 4513.

(12) See for example: Bokobza, L.; Pham-Van-Cang, C.; Giordano, C.; Monnerie, L.; Vandendriessche, J.; De Schryver, F. C. *Polymer* **1987,** *28,* 1876.

Monte Carlo Simulations of Linear and Cyclic Chains on Cubic and Quadratic Lattices

Johannes Reiter

Max-Planck-Institut für Polymerforschung, Postfach 3148, D-6500 Mainz, F.R.G.

Received October 6, 1989; Revised Manuscript Received February 2, 1990

ABSTRACT: A Monte Carlo algorithm for hypercubic lattices is investigated that combines end and kink reptations with local dynamic motions. It can be used for linear chains, for rings, and for the equilibration of the arms of star polymers. The algorithm fulfils the condition of detailed balance, and it is ergodic for a single linear chain. For the special case of a cyclic chain in two dimensions, a proof of ergodicity is also given. The statistical properties of the algorithm are discussed and, as examples, chain dimensions of linear and cyclic chains are computed.

Introduction

There are many algorithms known for the simulation of single macromolecules on a lattice.[1] Here we want to discuss yet another algorithm that might be used with advantage in some applications. The following motions are used: (1) inversion of L-structures, end turns, and (2) general reptation moves where a kink or end group is transported via a conformation change along the chain; see Figure 1. The latter motions include, as special cases, mere rotations of a kink, i.e., the well-known crankshaft motions, and reptations of the whole chain, i.e., slithering snake motions. Crankshaft motions combined with the inversion of L-structures and rotation of end segments are known not to be ergodic, and, in two dimensions, new bonds are generated only at the ends;[1] i.e., with these motions rings in two dimensions cannot be simulated. Because of the long-range transport of kinks, these disadvantages are avoided here. Long-range transport of kinks has already been used for dense systems at a volume fraction of unity in the collective motion algorithm.[2-4] Long-range motions have also been used by Skolnick et al. in simulations of lattice models for protein.[5] For single chains, removal and insertion of kinks has been introduced by Berg and Foerster[6] and de Carvalhalo et al.[7,8] in a grand canonical simulation of chains with varying length,[1,9] and the statistical properties of that algorithm have been discussed by Caracciolo and Sokal.[9] They demonstrated, for instance, that these motions are ergodic. As shown here

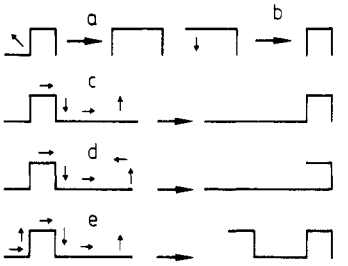

Figure 1. (a) L-inversion. (b) End turn. (c, d) General reptation motions are shown, where the chain changes conformation by a disappearance of a movable group, a slithering motion along the contour, and the formation of a movable group. (c) Kink–kink reptation. (d) Kink–end reptation. In the example shown, the end group takes on a kink conformation, but it could also take on other conformations. (e) End–kink reptation. Other motions not shown include end–end reptation, i.e., the familiar slithering snake motion, the rotation of a kink (crankshaft motion), and the conformation change of an end group containing two beads.

the removal and insertion of a kink (or an end group) can be combined into one step which is still ergodic.

The paper is organized as follows. First, the algorithm is discussed in detail and a proof for ergodicity for linear chains in any dimension and for cyclic chains in two dimensions is given. Then variants of the algorithm are discussed which might be used for denser systems. Finally, chain dimensions computed with the algorithm for linear

0024-9297/90/2223-3811$02.50/0

and cyclic chains are presented and the statistical properties of the algorithm are discussed.

The Algorithm

General Outline and Theory for Linear Chains. A Monte Carlo algorithm yields correct results if it fulfils the condition of detailed balance and if it is ergodic.[1,10,11] The former condition ensures that states are obtained with the correct probability corresponding to their free energy, and the second condition ensures that each state can be accessed. In the following, the general reptation algorithm is described for self-avoiding chains on a hypercubic lattice, and it is directly seen that it fulfils the condition of detailed balance. Then a proof for ergodicity is sketched which is inspired by the work of Madras and Sokal.[11]

On a hypercubic lattice in d dimensions a kink (or plaquette) consists of four segments and two segments are moved in a reptation. To simplify matters, in an end reptation two segments are moved also (Figure 1). A reptation motion consists of the disappearance of an end or a kink, which we call the shrink step, and of the formation of an end or a kink, which we call the growth step. The segments between the disappearing and the growing kink groups are moved along the contour of the chain. The elementary motion is a conformation change, but for the sake of the algorithm, it is convenient to assume that a movable group, i.e., a kink or an end, is cleaved at one position and attached at another position. The two lattice sites made free by the disappearing group are accessible to the newly forming group.

If excluded volume is not taken into account, there exist $(2d)^2$ different conformations for a two-site end group and $2d - 2$ different conformations for a kink that is attached to a given bond. The probabilities for selecting end motions and kink motions have to be calculated such that, in an athermal system, each conformation is equally likely. To ensure that this condition is fulfilled, the growth step may be done with one of the following methods. (1) First, a site is randomly selected for the attempt, where a kink growth on a given bond and an end growth at a given end are a priori equally likely. Then two random integers, i_{dir}, $1 \le i_{dir} \le 2d$, are chosen, which select one of the possible directions in the lattice. In the case of an end growth, they specify the growth directions for the first and the second bead, respectively. In the case of a kink growth, if the two random integers are equal and if the chosen direction is perpendicular to the bond selected in the growth attempt, a kink may be attached. (2) In a growth attempt, a given end is selected for an end growth attempt $2d$ times as often as a given bond for a kink growth attempt. An end growth attempt is then done as described above, but in a kink growth attempt, only one random integer has to be chosen. The second method is computationally more efficient and was used here. In the shrink step a given end or a given kink is removed with the same probability.

The algorithm then works as follows. If N is the number of beads of the chain, a random integer i_s, $1 \le i_s \le N - 1$, is chosen which specifies the bond for the shrink step: If $i_s = 1$ or $i_s = N - 1$, the first or the last two beads of the chain will be removed. If the bond i_s connects the two protruding beads of a kink, that kink is marked for the shrink. If no kink or end is found at bond i_s, the conformation is retained and recounted, and a new i_s is chosen. Otherwise, a second random integer i_g, $1 \le i_g \le N - 1 + 4d$, is chosen which specifies at which position a growth is attempted. If i_g falls in the range $[1, N - 1]$, a kink growth at the respective bond is attempted, if $N \le i_g \le N - 1 + 2d$, an end growth at the first chain end is attempted, and if $i_g \ge N + 2d$, an end growth at the other chain end is attempted. The growth step is successful if there is no overlap with other beads (thereby taking the excluded volume into account) and if the bond on which the growth step is attempted had not been removed by the shrink step. It is possible, however, to reattach a group to its old site in a different conformation. If both the shrink step and the growth step are successful, the reptation is performed in the athermal case. If the growth step is unsuccessful, the original conformation is retained and recounted and the procedure starts anew with selecting another i_s. In the case of semiflexible chains or if the interaction with empty sites (solvent) is different from self-interaction, a successful reptation has yet to pass the standard Metropolis test. If it fails, the old conformation is retained and recounted. End turns and L-inversions may be done as described in the literature. See Kremer and Binder[1] for references. The implementation of the algorithm is described in the Appendix.

To prove ergodicity it is sufficient to show that a straight rod conformation arbitrarily oriented in space may be reached starting from any chain conformation. First, the smallest hypercube is found that encloses the chain. In every surface of the hypercube there must be at least one end segment or one bond. If an end segment lies in a surface of the hypercube, with end–end reptations perpendicular to the surface a rod conformation can trivially be reached. If no end segment is in the surface of the hypercube, one bond is selected and kink growths on that bond perpendicular to the surface into the free space outside of the cube are combined with end shrinks. With these end–kink reptations, a U-shaped structure can be reached (the U might be asymmetrical). Then one of the two chain ends is selected randomly and with end–end reptations a straight rod conformation can again easily be reached. If we have a straight rod conformation, with end–end reptations it can be oriented arbitrarily in space. If we move the end segments always by two beads, there is an additional problem however. To see this let us calculate a parity $(-1)^{x1+...+xn}$ for each lattice site $(x1...xn)$. The parity can take on values of +1 and −1, and a site with a given parity has nearest neighbors of the opposite parity. It is clear that end segments will never change parity. This is no practical consequence in an isotropic space, however, since the motions could be used together with a translation of the whole chain by one lattice site which would change the parity of both ends. Such a motion does not change any of the static quantities that are usually measured in a Monte Carlo simulation, and it can therefore be safely omitted. In a nonisotropic space, for instance, near a surface with a position-dependent interaction to chain ends, the nonergodicity would be a problem. In the case of a linear chain one site reptation can of course always be used. One could also simultaneously elongate or cleave one segment from each end, instead of doing two-site end motions. In summary, we can conclude that general reptation motions are ergodic for a single linear chain. Figure 2 shows an example of a chain conformation that could not be changed with local dynamic motions and slithering snake motions. With general reptations the chain can still be equilibrated however.

Kink–kink reptations and end–kink reptations can of course also be used for the equilibration of the arms of a star polymer. Static properties for stars may be computed even if the central monomer rests immobile. For a suitable conformation of the center, it is in principle also possible to move the central monomer if general reptations on two

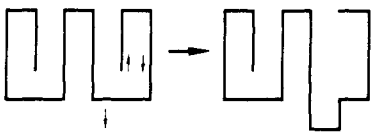

Figure 2. A linear chain is shown in a conformation where both ends are trapped. With slithering snake motions, end turns, L-inversions, and crankshaft motions the chain cannot be moved in two dimensions. However, end-kink reptations are still possible. Judging from that motion, the algorithm might be related to the sidewinder snake.

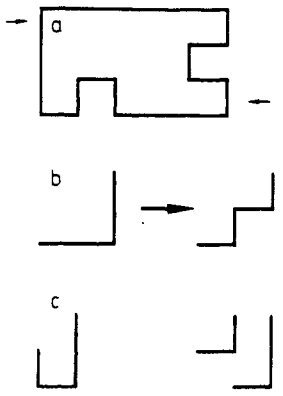

Figure 3. (a) A cyclic polymer in two dimensions is shown. The arrows point out the highest bond on the left and the lowest bond on the right that lie in the enclosing rectangle discussed in the text. (b) Shown is an L-structure in the conformation and the desired inversion direction discussed in the text. (c) An L-structure as shown in (b) cannot be inverted if it is part of a kink (left) or if the inversion point is blocked by a parallel L-structure (right).

(or more) arms are done concomitantly. To ensure that the detailed balance condition is fulfilled, such motions have to be selected with care. In this work we do not discuss these motions, however, but restrict ourselves to the basic motions that may be used on one arm.

Cyclic Chains. The algorithm is simpler for rings because i_s and i_g are now chosen from the interval [1, N] and correspond directly to a bond on which a kink-shrink step. or a kink-growth step is attempted. However, for cyclic polymers kink–kink reptations alone are not ergodic, because for any conformation without kinks, no motion is possible.

As we now show for cyclic polymers on a quadratic lattice, a combination of kink–kink reptations and inversion of L-structures is ergodic. Specifically we will show that with these motions any arbitrary conformation of a cyclic polymer of length N can be brought into a rectangular conformation of size 2 times $N/2$ (N is always even for a cylic chain on a hypercubic lattice). To this end we find the smallest rectangle that encloses the chain. Each side of the rectangle must contain at least one bond. We arbitrarily pick on the left side of the rectangle the uppermost bond and attempt to grow a kink on it toward the outside of the enclosing rectangle; see also Figure 3a. By the following procedure we can always find a kink that can be removed to enable the growth. We select the lowest bond that lies in the right hand side of the enclosing rectangle (Figure 3a). It is clear that the bond is part of a straight stretch of the polymer that lies in the right side of the enclosing rectangle and that is bordered at its lowest bead and its highest bead by bonds to the left. With L-inversions from below (Figure 3b), in principle a kink with a removable bond in the right side of the enclosing rectangle can be created which makes the kink–kink reptation possible. We have now to investigate the case where the L-inversion is not feasible because the inner site is taken by another bead of the polymer. As illustrated in Figure 3c, there are only two possible configurations at this point. The lower bead may form with the bead on the left the removable part of a kink. We are therefore finished in this case. The other case is that the inner site is blocked by another L-structure. Then we attempt to invert both L-structures simultaneously. For the inner L-structure the same considerations hold as for the outer: it can be inverted, it is part of a kink that may be removed, or it is blocked by another L-structure. We continue until either the stack of L-structures can be inverted returning to the lowest bond on the right side of the enclosing rectangle, or if a kink is found. Either of the two possibilities happens at the latest when the innermost L-structure touches with its left bead the left side of the enclosing rectangle. At this point no further L-structure can block the site of inversion because its left side would be outside of the enclosing rectangle. We conclude that a kink–kink reptation is always possible. The bond removed from the kink thus found is not identical with the bond on which the growth step takes place. Therefore, at the latest after $N/2$ kink–kink reptations done as described above, we reach the desired conformation of the polymer.

For three dimensions one would have to show that any cyclic polymer without knot can take on the conformation just described. (For polymer with at least one knot other conformations have to be used of course.) We do not attempt such a proof here, but the simulation results shown below indicate that the algorithm can be used also in three dimensions.

Variants of the Algorithm for Multichain Systems. While we will not discuss simulation of multichain systems in this work, for future reference we want to show that other versions of the algorithm that might be more suitable for dense systems are also ergodic. So far, a reptation move has been successful if a successful shrink step was combined with a successful growth step, where the vacancy created by the shrink step is available for the growth step. In dense systems with few vacancies it is more efficient to move vacancies instead of chains,[20] and we consider only motions where the vacancy created by the shrink step is *not* available for the growth step. Such a motion always amounts to a transport of a two-site vacancy. To distinguish the two techniques by name we call the usual one shrink–growth and the present one growth–shrink, although the order of selection of the steps may be done exactly as described above. The growth–shrink motions are a subset of the shrink–growth motions, and for any successful motion the reverse motion is always possible. The growth–shrink algorithm is therefore also detailed balanced. The only difference is that reptations where the growing and the shrinking movable groups share lattice sites are no longer possible in the growth–shrink version. Such reptations, however, indirectly take place as a combination of other reptations, and the algorithm is still ergodic. In fact, the proof for ergodicity given above utilizes only reptations that are in accordance with the growth–shrink version of the algorithm.

Finally, we want to introduce still another variant of the growth–shrink version of the algorithm where first the growth step is attempted and it is performed if possible. Then starting from the newly formed group the *nearest* kink or end either on the left or on the right is removed. If a kink is formed in the growth step, it is decided randomly if the left or the right neighbor is removed. After an end growth there is of course only one nearest neighbor. To fulfil the condition of detailed balance, in the growth step, kink growths have to be attempted twice as often as

described above. This algorithm is also ergodic for a single chain because if a kink or end can be found as outlined above, there must be a nearest-neighbor kink or end. The collective motion algorithm which works at a volume fraction of unity uses similar motions.[4]

Results

Linear Chains. Initially, the chain is prepared in a straight rod conformation and is then equilibrated for at least 20 integrated autocorrelation times (see below). To check if the algorithm yields correct chain dimensions, we compare our data to those obtained by exact enumerations in two dimensions. The results for the chain lengths $N = 16$ and $N = 21$ have been used by Madras and Sokal[11] to assess the accuracy of their computations performed with the pivot algorithm, and we also use these data. Our values for the mean square end-to-end distance, $\langle R^2 \rangle$, and the mean square gyration radius, $\langle s^2 \rangle$, are as follows: $N = 16$, $\langle R^2 \rangle = 47.23 \pm 0.04$ (47.2177), $\langle s^2 \rangle = 6.787 \pm 0.007$ (6.7843); $N = 21$, $\langle R^2 \rangle = 72.05 \pm 0.05$, 72.08 ± 0.06, 72.08 ± 0.018 (72.0765); $\langle s^2 \rangle = 10.235 \pm 0.01$, 10.246 ± 0.014; 10.245 ± 0.012 (10.2477 ± 0.0100). The first value is always calculated with the shrink–growth version of the algorithm. For $N = 21$ the second value is calculated with the growth–shrink version and the third value with the growth–shrink version where the movable group nearest to the newly grown group is selected for the shrink step. In parentheses the exact values as quoted by Madras and Sokal[11] are shown in the first three cases, and in the last case their value obtained with the pivot algorithm is used. Note that we use the number of beads for chain length, while these authors use the number of bonds. The error bars which give 95% confidence limits were calculated in the standard manner by determining the autocorrelation functions.[10,11] We used more than 10^5 independent chain conformations to calculate the chain dimensions. The results are the same irrespective whether L-turns and end turns are used or not. All further computations are done with the shrink–growth version of the algorithm.

To estimate how the computational expense increases with chain length, we measured the autocorrelation times for root mean square (rms) end-to-end distance and gyration radius for different lengths. For details on how the correlation function is estimated, see Madras and Sokal.[11] We simulated chains in two and three dimensions with the method described here, where the ratio of reptation moves to L-inversions and end turns was adjusted to about unity. We also simulated chains with the slithering snake method combined with local dynamic moves, i.e., end-to-end reptations, crankshaft motions, L-inversions, and end-turns, which is identical with the algorithm used here when long-range kink motions are omitted. The results are shown in Figure 4. In three dimensions for both methods the integrated autocorrelation times scale with about the 2.9th power of chain length. For two dimensions the exponent is somewhat larger (Figure 4). There is no significant difference between the two methods within our accuracy.

It is known that for the slithering snake algorithm, the autocorrelation time increases as about the square of chain length, while for algorithms that employ only local moves, the exponent is about $2(1 + \nu)$, where ν is the scaling exponent of the rms end-to-end distances, which is ca. 0.59 in three dimensions and 0.75 in two dimensions.[1] Note that in our simulations the ratio of end reptations compared to kink reptations and crankshaft motions decreases with chain length, N, as about $1/N$; see also the algorithm section. When only slithering snake motions

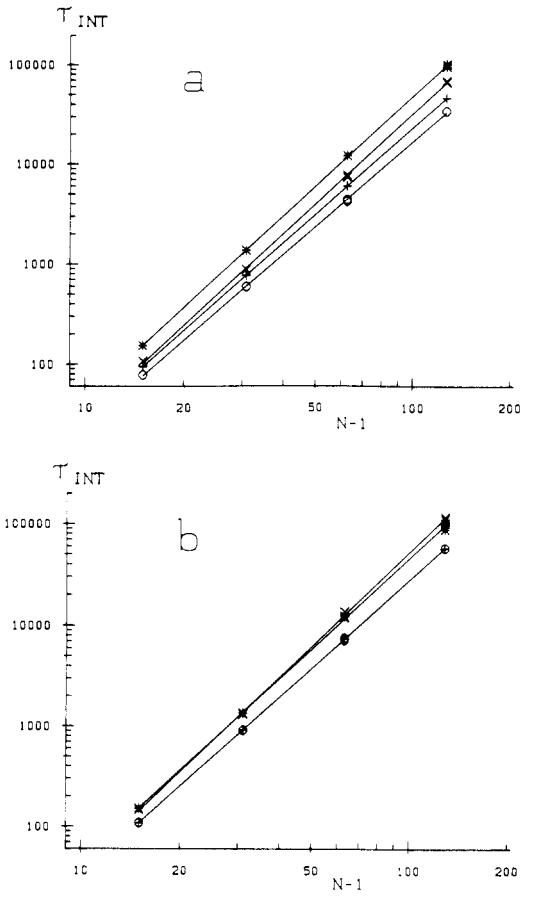

Figure 4. Integrated autocorrelation times, τ_{INT}, are shown as a function of number of bonds, $N - 1$, for the rms end-to-end distance (a) and the radius of gyration (b). The data are fitted to the equation $\tau_{INT} = N^{\gamma}$. (×, ○) Slithering snake motions combined with local dynamic motions, $d = 2$ and $d = 3$, respectively, where d is the spatial dimension; i.e., $d = 2$ stands for the quadratic and $d = 3$ for the cubic lattice. (*, +) Present algorithm, $d = 2$ and $d = 3$, respectively. The exponents in (a) are (×) 3.05, (○) 2.86, (*) 3.05, and (+) 2.92 and those in (b) are (×) 3.14, (○) 2.95, (*) 3.03, and (+) 2.95. The relative error as estimated from the fit is 3% or smaller.

are considered, the autocorrelation time should therefore increase as the third power of chain length. We indeed find for the slithering snake algorithm combined with local dynamic motions an exponent of ca. 3. Thus, it can be concluded that the slithering snake motions dominate the scaling. The general reptation algorithm yields approximately the same exponent for linear chains and, as shown below, for cyclic chains as well.

The integrated autocorrelation times give the time in attempted moves required to obtain an independent chain. Since the acceptance fraction of the motions and the ratio of standard deviation and measured quantity changes only by a few percent for the chain lengths used in our simulations (partly due to the changing fraction of end motions and inner chain motions), the computational expense of the simulation as a function of chain length is largely determined by the behavior of the integrated autocorrelation times.

The scaling of end-to-end distance and radius of gyration with chain length is in agreement with published results. We do not show these data because very accurate estimations already exist for a large range of chain lengths.[1]

Cyclic Chains. We have simulated rings on quadratic and cubic lattices (Figure 5a). The rms radius of gyration of the chain and the rms end-to-end distance of half of the chain ($N/2$ bonds, where N is the size of the ring) scale

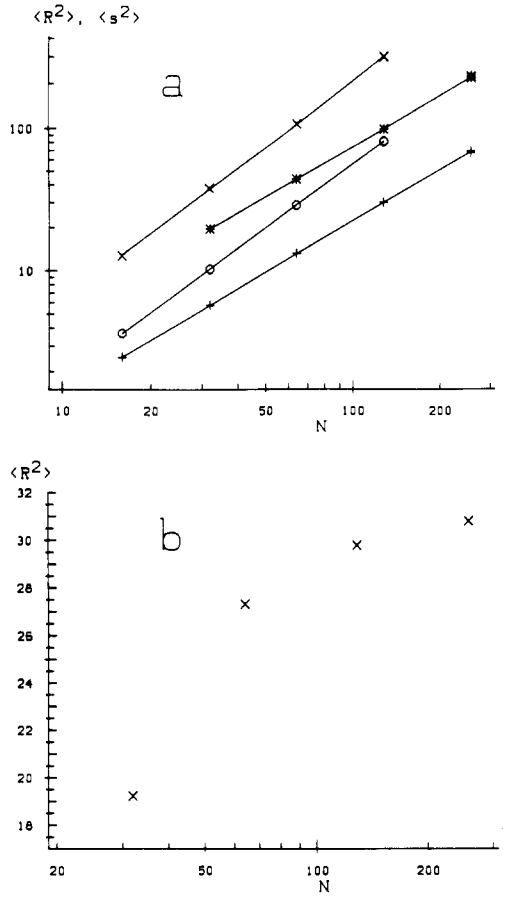

Figure 5. (a) Average square of the radius of gyration of the ring, $\langle s^2 \rangle$, and of the end-to-end distance of half of the ring ($N/2$ bonds), $\langle R^2 \rangle$, is plotted versus the ring size N for quadratic and cubic lattices. The lines are fits to the scaling relation $\langle R^2 \rangle, \langle s^2 \rangle \sim N^{2\nu}$. (×) $d = 2$, $\langle R^2 \rangle$, $\nu = 0.76 \pm 0.01$; (○) $d = 2$, $\langle s^2 \rangle$, $\nu = 0.747 \pm 0.003$; (*) $d = 3$, $\langle R^2 \rangle$, $\nu = 0.587 \pm 0.006$; (+) $d = 3$, $\langle s^2 \rangle$, $\nu = 0.597 \pm 0.003$. (b) The mean square end-to-end distance of a segment of 16 beads (15 bonds) taken from rings of different sizes is plotted against the ring size N.

with the chain length with an exponent of ca. 0.75–0.76 in two dimensions and 0.58–0.60 in three dimensions, which is, within these error limits, in agreement with scaling exponents determined for linear chains. Baumgärtner,[12] using a dynamic Monte Carlo method, investigated the scaling of freely jointed rings in the range from $N = 80$ to $N = 320$ in three dimensions and found a similar exponent. In a series of letters[13-16] Bishop and Michels investigated freely jointed rings with Brownian dynamics. For rings in the range from $N = 16$ to $N = 64$, they found slightly larger exponents, i.e., 0.77 and 0.625 in two and three dimensions, respectively. Probably for these off-lattice simulations longer chains are needed to reach the asymptotic regime. Other work on rings is discussed in ref 12–16.

In Figure 5b, the rms end-to-end distance of a sequence of 16 beads taken from rings of different sizes is shown. To understand these data we have to recall the properties of a random walk which returns to the origin,[17] i.e., an ideal ring, and an unrestricted random walk, i.e., an ideal chain. The rms end-to-end distance of an unrestricted random walk of n steps is proportional to $n^{1/2}$. Now lets consider an N-step random walk that returns to the origin after these N steps and take a sub-sequence of n steps from it, $n < N$. The rms distance of such a sub-sequence is smaller than that of the unrestricted walk. It calculates as $(n(N - n)/N)^{1/2}$. Therefore the closer n is in size to N, the more contracted is the subwalk. In the real ring, the situation is more complicated of course because the contraction is

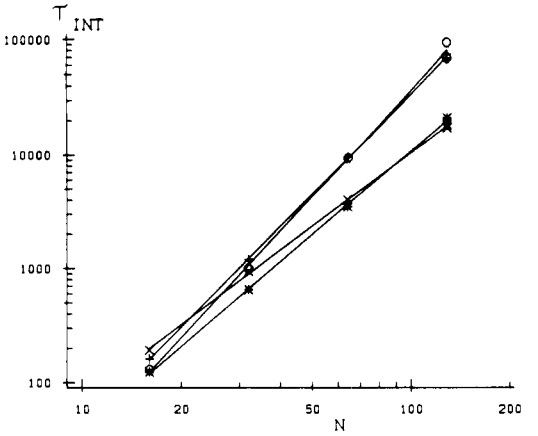

Figure 6. For ring polymers on quadratic and cubic lattices integrated autocorrelation times, τ_{INT}, are shown as a function of ring size, N, for the rms end-to-end distance of half of the ring ($N/2$ bonds) and the rms radius of gyration of the ring. The data are fitted to the equation $\tau_{INT} = N^{\gamma}$ (×) $d = 2$, $\langle R^2 \rangle$, $\gamma = 2.17$; (○) $d = 2$, $\langle s^2 \rangle$, $\gamma = 3.12$; (*) $d = 3$, $\langle R^2 \rangle$, $\gamma = 2.47$; (+) $d = 3$, $\langle s^2 \rangle$, $\gamma = 2.94$. The relative errors determined from the fit are ca. 3%.

counteracted by the excluded volume, but qualitatively the same effect is observed. The behavior shown in Figure 5b is then simply a consequence of the fact that rings are not self-similar; i.e., the dimension of a ring fragment strongly depends on the size of the ring.

In Figure 6 the dependence of the autocorrelation times on ring size is shown. For the radius of gyration the scaling exponent is again ca. 3 but for the end-to-end distance of half of the chain a smaller exponent of ca. 2.5 or smaller is determined. While we have no really satisfactory explanation, we believe that this discrepancy has to do with the fact that rings are not self-similar.

Conclusions

A general reptation algorithm has been discussed that allows the simulation of linear and cyclic chains and the arms of star polymers on a lattice. For a single linear chain the algorithm fulfils the condition of detailed balance and it is ergodic. The calculated chain dimensions compare well with literature data. For a single cyclic chain, ergodicity has been proved only for two dimensions, but the method is probably well suited also for three dimensions. The scaling of chain dimensions with chain length of cyclic chains is found to be very similar to that of linear chains.

The algorithm has been described for the special case of a hypercubic lattice, but it could be used for other lattices as well. On triangular and fcc lattices kinks consist of three sites, one of which is transportable.[18] Hence the parity problem of the hypercubic lattice does not arise and no other motions are required even for cyclic polymers. On a diamond lattice a kink consists of five sites, three of which are transportable. Similar to the case of a hypercubic lattice, in general the reptation motions should be combined with other local motions. In the one-site bond fluctuation model[19] kinks consist of three beads, one of which is transportable.

The computational expense as measured by the autocorrelation functions increases as about the third power of chain length. The exponent appears to be slightly larger in two dimensions. We found very similar exponents for slithering snake motions combined with crankshaft motions, L- and end turns. While that algorithm is not ergodic and ours is, this difference seems not to matter for single linear chains. However, for the conventional algorithm the computational expense increases strongly

with volume fraction at densities near unity.[1] The collective motion algorithm which is used at a volume fraction of unity also employs the general reptations. It may be that the latter algorithm owes its success to the ergodicity of these motions. Future work should address this question.

Acknowledgment. I thank K. Binder, T. Edling, E. W. Fischer, S. Geyler, K. Kremer (who also pointed out ref 18), and T. Pakula for discussions.

Appendix

Implementation of the Algorithm. Because in the simulation formally two beads are cleaved from one part of the chain and attached at another position, the order of beads gets scrambled and the chain cannot be stored simply as a linear function of position. Instead it is stored as a linked list where in array ch the beads are stored in arbitrary order and in two separate arrays of the same length and order a code is stored that specifies how to find from the respective bead the next and the previous bead position in the chain. That code also identifies chain ends. The lattice is stored as a linear array lt where the array index is a one-to-one function of the lattice coordinates. A free lattice site is encoded by zero, and, if the site is taken, the array entry specifies the position of the occupying bead in array ch. Conversly, in array ch the index of the lattice site is stored such that the following identity holds: $i_{pos} = lt(ch(i_{pos}))$. As described in the main text, random integers i_s and i_g are chosen. From array ch the lattice positions of these beads are found and with the help of the other arrays it can easily be checked if the reptation is possible. (We use the convention that the first bead identifies the bond; i.e., the last bead in a linear chain does not correspond to a bond. As outlined in the description of the algorithm, i_s should really be taken from the interval $[1, N - 1]$; however, since we do not know beforehand where the last bead in the chain is stored, we choose i_s from the interval $1...N$. If using the procedure given above we find the last bead in the chain, a new i_s is selected without recounting the conformation.) In the case of a linear polymer, the end positions are stored separately to find them faster if i_g is larger than the chain length. The computational effort per reptation attempt with such coding is independent of chain length. If i_s and i_g would correspond directly to positions in the chain, these would have to be found by scanning along the polymer from an initial bead and the average search time would be proportional to chain length. After a successful reptation the arrays have to be updated; that effort is again independent of chain length.

The inversion of L-structures and turning of end segments are done analogously: i.e., using array ch, a bead is randomly selected and if the movement is possible it is performed; otherwise the old conformation is retained and recounted. Again, with the coding done as described the effort per attempt is independent of chain length. For measuring gyration radii and other static properties, a sequentially ordered chain is needed that can easily be obtained by using one end position and the linked list. For rings one can start with any bead to restore an ordered sequence.

References and Notes

(1) Kremer, K.; Binder, K. *Comput. Phys. Rep.* **1988**, *7*, 259–310.
(2) Pakula, T. *Macromolecules* **1987**, *20*, 679–682.
(3) Pakula, T.; Geyler, S. *Macromolecules* **1987**, *20*, 2909–2914.
(4) Geyler, S.; Pakula, T.; Reiter, J. *J. Chem. Phys.* **1990**, *92*, 2676–2680.
(5) Skolnick, J.; Kolinski, A.; Yaris, R. *Biopolymers* **1989**, *28*, 1059–1096.
(6) Berg, B.; Foerster, D. *Phys. Lett.* **1981**, *106B*, 323–326.
(7) de Carvalhalo, C. A.; Caracciolo, S.; Fröhlich, J. *Nucl. Phys.* **1983**, *B215* [FS7], 209.
(8) de Carvalhalo, C. A.; Caracciolo, S. *J. Phys. (Paris)* **1983**, *44*, 323–331.
(9) Caracciolo, S.; Sokal, A. D. *J. Phys. A: Math. Gen.* **1986**, *19*, L797–L805.
(10) Binder, K. Introduction: Theory and Technical Aspects of Monte Carlo Simulations. In *Monte Carlo Methods in Statistical Physics*; Binder, K., Ed.; Springer-Verlag: Berlin, 1986; pp 1–45.
(11) Madras, N.; Sokal, A. D. *J. Stat. Phys.* **1988**, *50*, 109–186.
(12) Baumgärtner, A. *J. Chem. Phys.* **1982**, *76*, 4275–4280.
(13) Bishop, M.; Michels, J. P. J. *J. Chem. Phys.* **1985**, *82*, 1059–1061.
(14) Bishop, M.; Michels, J. P. J. *J. Chem. Phys.* **1985**, *83*, 4791–4792.
(15) Bishop, M.; Michels, J. P. J. *J. Chem. Phys.* **1985**, *84*, 444–446.
(16) Bishop, M.; Michels, J. P. J. *J. Chem. Phys.* **1986**, *85*, 1074–1076.
(17) See, e.g.: Burchard, W. Theory of Cyclic Macromolecules. In *Cyclic Polymers*; Semlyen, J. A., Ed.; Elsevier: London, 1986; p 47.
(18) For these lattices an algorithm that is similar to ours has recently been described: Murat, M.; Witten, T. *Macromolecules* **1990**, 23, 520–527.
(19) Carmesin, I.; Kremer, K. *Macromolecules* **1988**, *21*, 2819–2823.
(20) De Vos, E.; Bellemans, A. *Macromolecules* **1975**, *8*, 651–655.