Journal of Statistical Physics, Vol. 100, Nos. 5/6, 2000

Bilocal Dynamics for Self-Avoiding Walks

Sergio Caracciolo,[1] Maria Serena Causo,[2,5] Giovanni Ferraro,[3] Mauro Papinutto,[4] and Andrea Pelissetto[3]

Received December 3, 1999; final April 24, 2000

We introduce several bilocal algorithms for lattice self-avoiding walks. We discuss their ergodicity in different confined geometries, for instance in strips and in slabs. A short discussion of the dynamical properties in the absence of interactions is given.

KEY WORDS: Self-avoiding walks; Monte Carlo algorithm; ergodicity; polymer; dynamic critical behaviour.

1. INTRODUCTION

The lattice self-avoiding walk (SAW) is a well-known model for the critical behaviour of a homopolymer in a solvent[1,2] and it has been extensively used in the study of several properties of heteropolymers.[3,4] Experiments are usually performed using monodisperse solutions, and thus, extensive work has been done to devise Monte Carlo algorithms to simulate fixed-length SAWs. Historically, the earliest algorithms used a local dynamics:[5] at each step, a small part of the walk (usually 2-4 consecutive beads) was

[1] Scuola Normale Superiore and INFN–Sez. di Pisa, I-56100 Pisa, Italy; e-mail: Sergio.Caracciolo@sns.it.

[2] NIC, Forschungszentrum Jülich, D-52425 Jülich, Germany; e-mail: M.S.Causo@fz-juelich.de.

[3] Dip. di Fisica and INFN–Sez. di Roma I, Università di Roma I, I-00185 Roma, Italy; e-mail: Andrea.Pelissetto@roma1.infn.it.

[4] Dip. di Fisica and INFN–Sez. di Pisa, Università di Pisa, I-56100 Pisa, Italy; e-mail: papinutt@cibs.sns.it.

[5] Present address: Dept. of Chemistry, University of Toronto, Ontario, Canada.

0022-4715/00/0900-1111$18.00/0

modified. Although easy to implement, these algorithms suffer a very serious drawback: as shown by Madras and Sokal,[6] any local algorithm is not ergodic and simulations span only an exponentially small part of the phase space. A different algorithm was inspired by an attempt to model the true dynamics of the polymer in the solvent: the reptation algorithm.[7–10] However, it was soon realized[8, 9] that also this algorithm is not ergodic because of the possibility of configurations with trapped endpoints.[6] These ergodicity problems can be solved using chain-growth algorithms[12–14] or non-local algorithms.[15–19] In the absence of any interaction, non-local algorithms are very efficient. For instance, in the pivot algorithm[17] the autocorrelation time for global observables increases linearly with the number of steps N, which is the optimal behaviour since it takes a time of order N simply to write down the walk. On the other hand, they are not very useful in the presence of strong attraction or in finite geometries.[7]

In the presence of surfaces, non-local algorithms are not even ergodic in general. For instance, as we will show, the pivot algorithm is not ergodic in a strip. Moreover, even when they are ergodic, they are not suited for the study of surface transitions since non-local moves will generate new walks with large energy differences and, thus, they will be rejected. Therefore, the dynamics will be very slow.

Non-local algorithms are also not suited for the study of the behaviour of homopolymers in the collapsed phase and of heteropolymers near the folding temperature.[22–24] Indeed, typical configurations are compact and the probability of success of non-local moves is small. Most of the simulations[25–32] use local dynamics, which is expected to correctly describe the physical kinetics[8] of a polymer in a dilute solution in the absence of

[6] It should be noted that the ergodicity problem is less severe for reptation than for local algorithms. For instance, for reptation one can prove (see ref. 6 and Prop. 9.2.3 in ref. 11) that the fraction of walks belonging to the ergodicity class of the straight rod is larger than $\sim N^{-(\gamma-1)/2}$. On the other hand, for local algorithms, each ergodicity class contains only an exponentially small fraction of the walks.

[7] An exception is the class of moves introduced in ref. 20 that do not change the position of the beads but only the connectivity of the walk. However, these moves do not change global size observables and are of interest only for the study of maximally compact configurations. A general lower bound on the efficiency of non-local algorithms with a Metropolis test is given in ref. 21.

[8] Non-local algorithms have also been considered.[33–35] However, a non-local dynamics which involves rigid deformations of a large section of the polymer is unphysical, and therefore cannot give realistic results for the physical kinetics. Moreover non-local moves become irrelevant—they are never accepted—as the number of monomers increases.

hydrodynamic interactions.[36, 37] It is however important to stress that all these algorithms are not ergodic so that systematic deviations are expected.[9]

In this paper we wish to discuss a family of algorithms that use bilocal moves: a bilocal move alters at the same time two disjoint small groups of consecutive sites of the walk that may be very far away. Since a small number of beads is changed at each step, these algorithms should be reasonably efficient in the presence of interactions, and thus they can be used in the study of the collapsed phase and of the folding of heteropolymers. They generalize the reptation algorithm and use a more general class of moves that was introduced by Reiter.[38] Similar moves were introduced in ref. 39 and were applied to the study of ring polymers,[10] and in ref. 33.

We will study in detail the ergodicity of these algorithms and we will show that, with a proper choice of moves, they are ergodic even in some constrained geometries, e.g., in strips, slabs, and generalizations thereof. These results have been obtained for SAWs with nearest-neighbour jumps on a (hyper-)cubic lattice. However, they can be easily generalized to different lattice models, for instance to the protein model proposed in ref. 40.

The paper is organized as follows. In Section 2 we introduce several local and bilocal moves and we define three bilocal algorithms. In Section 3 we discuss their ergodicity, determining the minimal set of moves that make each algorithm ergodic. This is important in order to understand the dynamical behaviour. Indeed, algorithms that are ergodic only if rarely accepted or rarely proposed moves are included, are expected *a priori* to have a slow dynamics. Section 4 contains a detailed presentation of their implementation. In the last section we present a brief discussion of the

[9] Reference 28 claims that, at the presently investigated values of N, non-ergodicity effects should be small. A quantitative study of the Verdier–Stockmayer algorithm in two dimensions[5] was presented in ref. 6. The percentage of walks that do not belong to the ergodicity class of the straight rod is indeed small, precisely 0.0067%, 0.0061%, 0.0041% for $N = 11, 13, 15$. However, it should be noted that this percentage increases with N (each ergodicity class contains only an exponentially small fraction of the walks[6]) and that the walks that do not belong to the ergodicity class of the straight rod correspond to compact configurations: therefore, larger systematic deviations axe expected in the collapsed regime. If we indicate with $c_N(n)$ the number of walks with N steps and n nearest-neighbor contacts and with $d_N(n)$ the corresponding number of walks that do not belong to the ergodicity class of the straight rod, a good indication of the deviations expected in the presence of strong attraction is given by $R_N = d_N(n_{\max})/c_N(n_{\max})$ and $S_N(\varepsilon) = \sum_n e^{n\varepsilon} d_N(n)/\sum_n e^{n\varepsilon} c_N(n)$. Here $n_{\max}$ is the maximum number of possible contacts for a given N; clearly $R_N = S_N(\infty)$. For the Verdier–Stockmayer algorithm we considered above, we have $R_N = 3.2\%$, 1.4%, 5.8% for $N = 11, 13, 15$, and $S_N(1) = 0.20\%$, 0.20%, 0.16% for the same values of N. Clearly, the systematic error is not completely negligible in the compact regime.

[10] However, it should be noted that the algorithm of ref. 39 is not ergodic.

expected dynamic critical behaviour in the absence of interactions. A detailed numerical study will appear elsewhere.[41]

2. DEFINITION OF THE ALGORITHMS

In this paper we will consider SAWs with fixed number of steps N and free endpoints in finite geometries.

More precisely, we consider a d-dimensional hyper-cubic lattice, $d \geqslant 2$, and define the following set of lattice points: given an integer D such that $1 \leqslant D \leqslant d-1$, and $(d-D)$ positive integers $w_{D+1},..., w_d$, we define $\mathscr{C}_D(w_{D+1},..., w_d)$ as the set of lattice points $(n_1,..., n_d)$, $n_i \in Z$, such that $0 \leqslant n_i \leqslant w_i$, for $i = D+1,..., d$. We will call[11] $\mathscr{C}_D(w_{D+1},..., w_d)$ a D-dimensional *cylinder*. If $D = (d-1)$, we will speak of a strip if $d = 2$ and of a slab if $d = 3$. The number w_i will be called the *width* of the cylinder in the ith direction. Note that we will always assume $D \geqslant 1$, so that at least the first direction is infinite.

We will then consider SAWs of length N confined inside a cylinder. A SAW ω is a set of $N+1$ lattice points $\omega(0),..., \omega(N)$ such that: $\omega(i)$ and $\omega(i+1)$ are lattice nearest neighbours; $\omega(i) \neq \omega(j)$ for any $i \neq j$; $\omega(i) \in \mathscr{C}_D(w_{D+1},..., w_d)$ for all i. We define two different ensembles:

1. the ensemble $\mathscr{E}_{x,N}$ of SAWs of length N such that $\omega(0) = x$;
2. the ensemble $\mathscr{E}_N$ of SAWs of length N such that both endpoints can be anywhere in the cylinder.

In free space the two ensembles are equivalent as long as one is interested in properties of the walk itself. In confined geometries they are different since translation invariance is lost in $d-D$ directions. Both ensembles are of physical interest: in a slab one can study the statistical properties of polymers that can move freely between the confining surfaces, or one can determine the behaviour of polymers grafted at one of the boundaries.

We wish now to define some algorithms that sample these ensembles of walks. They use local and bilocal moves. A local move is one that alters only a few consecutive beads of the SAW, leaving the other sites unchanged. A bilocal move is instead one that alters two disjoint small groups of consecutive sites of the walk; these two groups may in general be very far from each other.

[11] In two dimensions, one often calls cylinder a strip with periodic boundary conditions. Note that the definition given here is different.

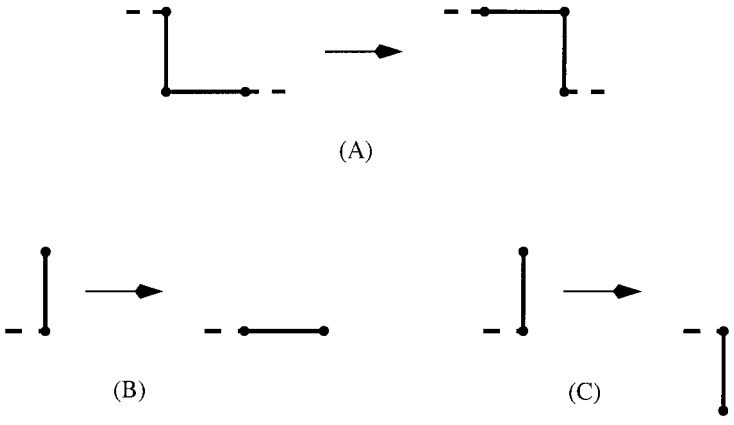

Fig. 1. All one-bead moves: (A) One-bead flip. (B) 90° end-bond rotation. (C) 180° end-bond rotation.

In our study we consider three types of local moves (see Figs. 1 and 2):

[L0] One-bead flips in which one *internal* bead (i.e., $\omega(i)$, $1 \leqslant i \leqslant N-1$) only is moved.

[L00] Kink rotations (also called crankshaft moves) in which a three-step kink is rotated. Note that 180° rotations are possible for all $d \geqslant 2$, while 90° rotations are defined only for $d \geqslant 3$.

[L1] End-bond rotations in which the last step of the walk is rotated. In ensemble $\mathscr{E}_N$ (but *not* in $\mathscr{E}_{x,N}$), the same move can also be applied to the first step of the walk.

We also introduce several types of bilocal moves:

[B22] Kink-transport moves in which a kink is cleaved from the walk and attached at a pair of neighbouring sites somewhere else along the walk (see Fig. 3); note that the new kink is allowed to occupy one or both of the sites abandoned by the old kink.

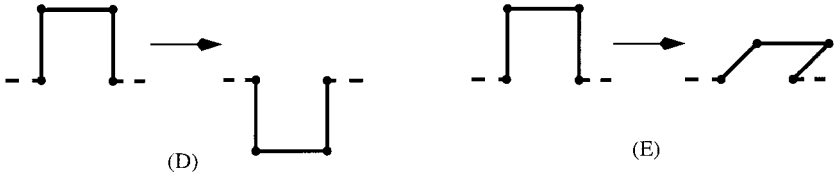

Fig. 2. Kink rotations or crankshaft moves.

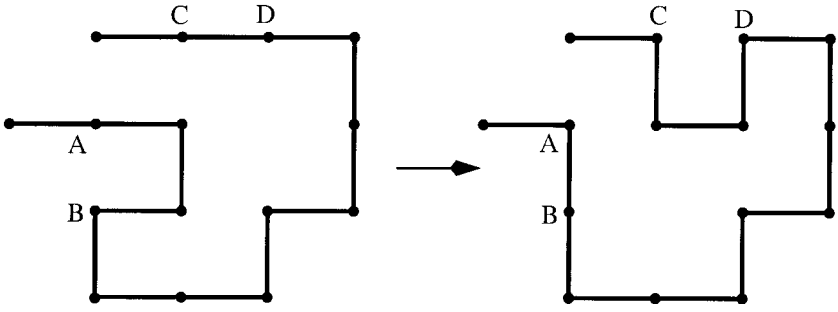

Fig. 3. The kink-transport move. A kink has been cleaved from AB and attached at CD. Note that the new kink is permitted to occupy one or both of the sites abandoned by the old kink.

[BKE] Kink-end and end-kink reptation moves (see Fig. 4). In the kink-end reptation move a kink is deleted at one location along the walk and two new bonds are appended in arbitrary directions at the free endpoint of the walk. Viceversa, an end-kink reptation move consists in deleting two bonds from the end of the walk and in inserting a kink, in arbitrary orientation, at some location along the walk. In ensemble $\mathscr{E}_N$, the same move can also be applied to the first step of the walk.

[BEE] Reptation move (see Fig. 5) in which one bond is deleted from one end of the walk and a new bond is appended in arbitrary direction at the other end. The move is allowed in ensemble $\mathscr{E}_N$ but not in ensemble $\mathscr{E}_{x,N}$.

We wish now to define algorithms made up with the moves we have presented above and that are ergodic. As shown by Madras and Sokal,[6] there exists no ergodic algorithm made up of local moves. It is therefore necessary to add some bilocal moves to obtain ergodicity. The oldest bilocal algorithm is the reptation algorithm, which uses only the moves BEE. As it was soon realized, it is not ergodic due to the presence of walks with

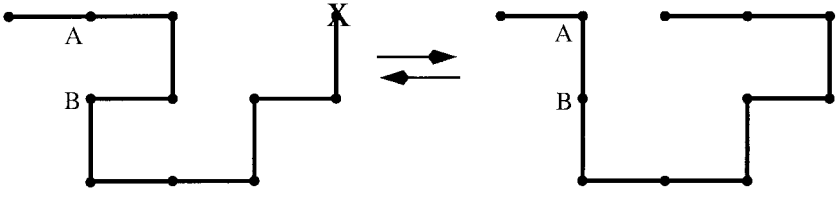

Fig. 4. The kink-end reptation ($\rightarrow$) and end-kink reptation ($\leftarrow$) moves. In ($\rightarrow$), a kink has been cleaved from AB and two new steps have been attached at the end marked X. Note that the new end steps are permitted to occupy one or both of the sites abandoned by the kink.

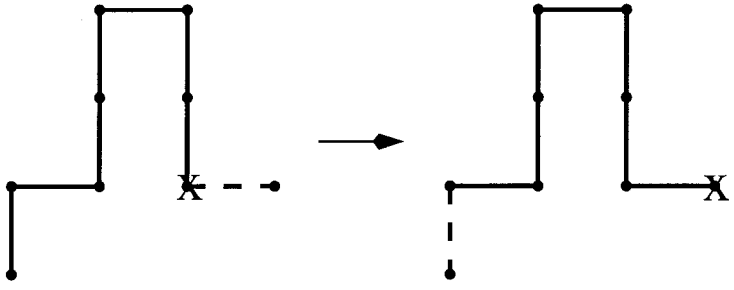

Fig. 5. The reptation move. The head of the walk is indicated by X. The dashed lines indicate the proposed new step and the abandoned old step.

trapped endpoints. Here we wish to define new bilocal algorithms. We first consider the ensemble $\mathscr{E}_{x,N}$. We discuss two algorithms:

• *Kink-kink bilocal algorithm.* It uses the local moves L0, L1, and the bilocal moves B22. The local move L1 is applied only to the last point of the walk, otherwise $\omega(0)$ would not be kept fixed. We will show that it is ergodic in two dimensions (under some technical conditions), and that it is not ergodic in three dimensions due to the possibility of knots. In higher dimensions its ergodicity is an open problem.

• *Kink-end reptation.* It uses the moves BKE applied to the last step of the walk. We will show that it is ergodic in a D-dimensional cylinder for $d \geqslant 3$. In two dimensions it is ergodic only in free space or in the presence of a single surface, i.e., for $D = 1$ and $w_2 = \infty$. There exists an extension that is ergodic in a two-dimensional strip: it uses the moves BKE applied to the last step of the walk and the local moves L0.

It is trivial to modify these algorithms so that the first point is not kept fixed. It is enough to apply L1 and BKE moves to the first step of the walk, too. However, these modifications are not ergodic in the ensemble $\mathscr{E}_N$: indeed, L1 and BKE moves never change the parity of the first point. If there is translation invariance in one infinite direction, this limitation is irrelevant: all walk properties can still be obtained correctly from the ensemble of walks in which the parity of $\omega(0)$ is fixed. However, this is not the case in the presence of random interactions, since translation invariance is completely lost. In order to sample the ensemble $\mathscr{E}_N$, we introduce a different algorithm:

• *Extended reptation.* It uses the moves L0, B22, BEE. We will show that it is ergodic in two dimensions. The ergodicity for $d \geqslant 3$ is an open problem. In the absence of a definite result, an ergodic extension in $d = 3$ can be obtained by adding BKE moves.

The ergodicity properties of these algorithms are proved in the next section. It should be noticed that we have not considered the local moves L00, which are not necessary for ergodicity, but that can be added to the moves L0 if one wishes.

The bilocal moves we have introduced above have already been discussed in the literature. Removals and insertions of kinks were introduced in refs. 42–45 in order to study cyclic polymers and polymers with fixed endpoints with varying length N. An ergodic algorithm was introduced by Reiter:[38] he considers moves L0, L1, B22, BEE, BKE and proves the ergodicity of the dynamics (his ergodicity proof requires only BKE and BEE moves) in free space. A similar algorithm was used in ref. 39 in a study of cyclic polymers, considering L0, L00, and BEE moves. In two dimensions the algorithm is ergodic even in a strip, as we shall show below, while in three dimensions it is not ergodic since it does not change the knot type of the loop. It is unknown if it is ergodic in a given knot class.

3. ERGODICITY

3.1. Definitions

In this section we will prove several ergodicity theorems for the algorithms we have introduced before.

We begin by introducing some definitions, following ref. 11. We make use of the following notations: We indicate by $\omega[i, j]$, $0 \leqslant i < j \leqslant N$, the subwalk connecting $\omega(i)$ to $\omega(j)$, i.e., the set of points of the walk ω: $\{\omega(k) : i \leqslant k \leqslant j\}$. The lattice link connecting $\omega(i)$ to $\omega(i+1)$, $0 \leqslant i \leqslant N-1$, is indicated by $\Delta\omega(i)$.

Definition 1. A subwalk $\omega[i, j]$, $(0 \leqslant i < j \leqslant N)$ is a C-turn of ω if $j - i \geqslant 3$, $\omega[i+1, j-1]$ lies on a straight line that is perpendicular to the steps $\Delta\omega(i)$ and $\Delta\omega(j-1)$ and $\Delta\omega(i) = -\Delta\omega(j-1)$.

The length of the C-turn is the length of the subwalk.

We say that a C-turn *belongs* to a line (or surface), or that a line *contains* a C-turn, if the segment $\omega[i+1, j-1]$ lies on the line (or surface).

Definition 2. A C-turn of ω, $\omega[i, j]$ is *obstructed* if there is a site of ω lying on the open line segment whose endpoints are $\omega(i)$ and $\omega(j)$. Otherwise it is *unobstructed*.

Definition 3. The *enveloping hyper-rectangle* $R[\omega]$ of the walk ω is the minimal lattice hyper-rectangle containing ω. A lattice hyper-rectangle

is the set of points $(x_1,..., x_d)$, such that $l_1 \leqslant x_1 \leqslant L_1$, $l_2 \leqslant x_2 \leqslant L_2,..., l_d \leqslant x_d \leqslant L_d$, for some integers $l_1, l_2,..., L_1, L_2,...$.

Definition 4. A walk ω is *directed* if there are no steps that have opposite directions.

Definition 5. A *tower* of links of height $h \geqslant 0$ is a subwalk $\omega[i, j]$ with $j-i=2h+1$, $0 \leqslant i < j \leqslant N$, such that $\omega[i, i+h]$ and $\omega[i+1+h, j]$ are segments and $\omega(i)$ and $\omega(j)$ are lattice nearest neighbours.

We call the lattice link l connecting $\omega(i)$ and $\omega(j)$ the *base* of the tower. We denote the tower as $T(l, h)$.

Moreover, we will say that a tower is *parallel* to a given line if the segments $\omega[i, i+h]$ and $\omega[i+1+h, j]$ are parallel to this line.

Definition 6. Given a walk ω and a tower $T(l, h)$, l connecting $\omega(i)$ and $\omega(j)$, we define the *quotient walk* $\omega/T(l, h)$ as the walk with sites $\omega(0),..., \omega(i), \omega(j),..., \omega(N)$.

Definition 7. Given a walk ω with a tower $T(l, h)$, let $\bar{\omega} = \omega/T(l, h)$ be the quotient walk. If $\bar{\omega}$ is directed, ω is said to be *quotient-directed.*

For the two dimensional proofs we will make extensive use of the following theorem due to Madras and reported in ref. 11, Theorem 9.7.2, p. 356:

Theorem 1. In two dimensions, if a walk ω has at least one C-turn, then ω has an unobstructed C-turn.

3.2. Ergodicity Properties of the Pivot Algorithm in a Confined Domain

In this section we want to discuss the ergodicity properties of the pivot algorithm. Following ref. 17, it is possible to prove that the algorithm is ergodic in the presence of a single confining surface. More precisely, the following theorem holds:

Theorem 2. Consider a $(d-1)$-dimensional cylinder in d dimensions. The pivot algorithm is ergodic in $\mathscr{E}_{x, N}$ if $N < \max(\omega_d(0), w_d - \omega_d(0))$, where $\omega_d(0)$ is the dth component of $\omega(0)$ and N is the number of steps of the walk.

However, the algorithm is not expected to be ergodic in more constrained geometries. We will now show that the pivot algorithm is not ergodic in a two-dimensional strip of width w, for $N > (w+1)^2$. Indeed, pick a bead $\omega(i)$ and let d be its distance from the boundary $y=0$. Then, consider the reflections with respect to the diagonals and the $\pm 90^\circ$ rotations. These are the only transformations that change the number of links that are oriented in the $\pm x$ and in the $\pm y$ directions. It is easy to see that these moves are successful only if either $-d \leqslant \omega(j)_x - \omega(i)_x \leqslant w-d$ or $d-w \leqslant \omega(j)_x - \omega(i)_x \leqslant d$ for all $j > i$. But this cannot be true if $N-i > (w+1)^2$. Consider now the subwalk $\Omega = \omega[0, N-(w+1)^2]$. The previous argument shows that the number of links belonging to Ω that are directed in the $\pm y$ or in the $\pm x$ is fixed. Thus, the algorithm is not ergodic. We believe, although we have not been able to prove, that the algorithm is also not ergodic in a three-dimensional slab.

3.3. Ergodicity of the Kink-Kink Bilocal Algorithm

We will now prove ergodicity in $\mathscr{E}_{x,N}$ of the kink-kink bilocal algorithm in two dimensions in a strip of width $w \equiv w_2$. To simplify writing the walks, we indicate by N and S the positive and negative y-direction, and by E and W the positive and negative x-direction. Let us begin by proving the following lemmas. In all cases we assume $d=2$.

Lemma 1. Consider a directed walk ω and suppose that the distance between $\omega(0)$ and at least one boundary of the strip is larger than or equal to 2. Then, ω can be reduced to any given rod using the kink-kink bilocal algorithm.

Proof. Using L0 and L1 moves, it is trivial to show that it is possible to reduce the walk either to W^N or to E^N. We will now show that it is possible to deform one rod into the other. If $N \leqslant 2$, the procedure is trivial and thus we will assume $N \geqslant 3$.

Using L0 and L1 moves, we can deform E^N as follows:

$$E^N \to E^{N-1}N \to NE^{N-1} \to NE^{N-2}N \to N^2E^{N-2} \tag{1}$$

where we have assumed that the distance between $\omega(0)$ and the upper boundary of the strip is at least 2. If this is not the case, by hypothesis, the distance between $\omega(0)$ and the lower boundary of the strip is at least 2, so that the rod E^N can be analogously reduced to S^2E^{N-2}. The steps we

will present below should then be changed replacing N by S. Then, by repeatedly performing the following sequence ($p \geqslant 2$)

$$W^k N^2 E^p \to W^k N^2 E^{p-1} S \to W^k N^2 E S E^{p-2} \to W^{k+1} N E^p$$
$$\to W^{k+1} N E^{p-1} N \to W^{k+1} N^2 E^{p-1} \tag{2}$$

we deform the walk into $W^{N-3} N^2 E$. Finally

$$W^{N-3} N^2 E \to W^{N-3} N^2 W \to W^{N-2} N^2 \to W^N \quad \blacksquare \tag{3}$$

The proof of lemma 1 requires a technical hypothesis about the distance between $\omega(0)$ and the boundaries of the strip. For $w > 2$, this condition is always satisfied. In the following we will consider walks such that $\omega(0)$ is not a nearest neighbour of a boundary of the strip. In this particular case, the hypothesis of Lemma 1 is also satisfied for $w = 2$, and thus for all $w \geqslant 2$.

Lemma 2. Consider a quotient-directed walk ω in a two-dimensional strip of width $w \geqslant 2$ and suppose that $\omega(0)$ is not a nearest neighbour of a boundary of the strip. Then, ω can be reduced to any given rod using the kink-kink bilocal algorithm.

Proof. Consider $\bar{\omega} = \omega / T(l, h)$, and assume that all steps of $\bar{\omega}$ are directed in the N, E directions. It is immediate to verify that, by using L0 and L1 moves, one can modify the walk obtaining one of these four possibilities:

(a) $N^{k_1} E^{k_2} N^h E S^h E^{k_3}$;

(b) $N^{k_1} E^{k_2} S^h E N^h E^{k_3}$;

(c) $E^{k_1} N^{k_2} E^h N W^h N^p E^{k_3}$;

(d) $E^{k_1} N^{k_2} W^h N E^h N^p E^{k_3}$.

We assume $h > 0$; otherwise, the walk is already directed and it can be reduced to any rod by Lemma 1. In case (a), we can use L0 moves to modify the walk into $N^{k_1+h} E^{k_2+k_3+1} S^h$, and then, combining L0 and L1 moves, into $N^{k_1+h} E^{h+k_2+k_3+1}$. The new walk is directed and, by Lemma 1, it can be reduced to any given rod.

In case (b), using L0 and L1 moves, we modify the walk as follows:

$$\omega \to N^{k_1} E^{k_2} S^h E^{k_3+1} N^h \to N^{k_1} E^{k_2} S^h E^{k_3+1} N^{h-1} E \to N^{k_1} E^{k_2} S^h E^{k_3+2} N^{h-1}$$
$$\to N^{k_1} E^{k_2} S^h E^{h+k_3+1} \to \cdots \to N^{k_1} E^{h+k_2+k_3+1} S^h \to N^{k_1} E^{h+k_2+k_3+1} S^{h-1} E$$
$$\to \cdots \to N^{k_1} E^{h+k_2+k_3+2} S^{h-1} \to \cdots \to N^{k_1} E^{2h+k_2+k_3+1} \tag{4}$$

which is directed. By Lemma 1, it can be reduced to any given rod.

Let us now consider case (c). If $p > 1$, using L0 and L1 moves, we can deform the walk into a new one with $p = 1$. If $p = 1$ and $k_3 > 0$, using B22 moves we obtain

$$\begin{aligned} E^{k_1}N^{k_2}E^hNW^hNE^{k_3} &\to E^{k_1}N^{k_2}E^{h+1}NW^hNE^{k_3-1} \\ &\to \cdots \to E^{k_1}N^{k_2}E^{h+k_3}NW^hN \end{aligned} \tag{5}$$

Then, using L0 and L1 moves, we can deform the walk as follows:

$$\begin{aligned} E^{k_1}N^{k_2}E^{h+k_3}NW^hN &\to E^{k_1}N^{k_2}E^{h+k_3}NW^{h+1} \\ &\to \cdots \to E^{k_1+k_3+h}N^{1+k_2}W^{h+1} \\ &\to E^{k_1+k_3+h}NW^{h+k_2+1} \end{aligned} \tag{6}$$

It is obvious that these last transformations can also be applied when $p = 0$ (in this case we have also $k_3 = 0$). Therefore, all walks can be transformed into new ones of the form E^pNW^q. If $q = 0$, the walk is directed, while for $q = 1$ we can transform it into E^pNE which is also directed. For $q \geqslant 2$, we transform the walk into $E^pN^2W^{q-1}$. Since $\omega(0)$ is not a nearest neighbour of the boundary, there is no obstruction to this transformation. If $q = 2$, by means of an L1 move, we obtain a directed walk. For $q \geqslant 3$, we apply repeatedly the following sequence

$$\begin{aligned} E^pN^2W^{q-1} &\to E^pN^2W^{q-2}S \to E^pN^2WSW^{q-3} \\ &\to E^{p+1}NW^{q-1} \to E^{p+1}N^2W^{q-2} \end{aligned} \tag{7}$$

obtaining a walk with $q = 2$ and then a directed walk. Using Lemma 1, the walk can be reduced to any rod.

Finally we consider case (d). If $k_2 = 0$, since $h > 0$, we have $k_1 = 0$. Then using L0 moves we can deform the walk into $W^hN^{p+1}E^{k_3+h}$. If $k_2 > 0$, using L0 moves we can modify the walk into $E^{k_1}NW^hN^{k_2+p}E^{h+k_3}$. Then, by applying repeatedly the transformation

$$\omega \to E^{k_1-1}NW^hNEN^{k_2+p-1}E^{k_3+h} \to E^{k_1-1}NW^hN^{k_2+p}E^{k_3+h+1} \tag{8}$$

we obtain a new walk of the form $NW^hN^{k_2+p}E^{k_1+k_3+h}$ and then $W^hN^{k_2+p+1}E^{k_1+k_3+h}$. Thus the walk can always be deformed into a new one of the form $W^pN^lE^q$, which can be reduced to a rod following the method applied to E^pNW^q. ▮

It should be noticed that the hypothesis of Lemma 2 is a necessary condition for its validity. Indeed, consider a walk with $\omega(0) = (0, 1)$,

$\omega(N-2)=(-1,0)$, $\omega(N-1)=(0,0)$, $\omega(N)=(1,0)$. This walk cannot be deformed into a rod, since the endpoint is frozen. This hypothesis is not required if one considers walks for which the first point is not fixed, allowing moves L1 also on the first step of the walk.

Lemma 3. Consider a walk ω in a two-dimensional strip of width $w \geqslant 2$. Then, ω can be reduced to a quotient-directed walk using the kink-kink bilocal algorithm.

Proof. Let us first introduce a few notations. To every walk ω (which is not a rod) with tower $T(l, h)$ we associate a triple $(\omega, T(l, h), t)$. In order to define t, set $\bar{\omega}=\omega/T(l, h)$ and consider $R[\bar{\omega}]$. Define R_1 and R_2 as the two sides of $R[\bar{\omega}]$ which are perpendicular to the boundary of the strip. Let us introduce coordinates so that the x-axis is along the strip. If R_1 and R_2 have equations $x=x_1$ and $x=x_2$ respectively, and (x_0, y_0), (x_N, y_N) are the coordinates of $\omega(0)$ and $\omega(N)$ respectively, it is not restrictive to assume $|x_1-x_N| \leqslant |x_1-x_0|$ and $x_1 \leqslant x_2$. If $|x_1-x_N| \neq 0$, set $t=0$. Otherwise, find the smallest $\bar{x}$ such that the line $x=\bar{x}$ contains a step of $\bar{\omega}$ or the starting point $\omega(0)$. Then, set $t=\bar{x}-x_1$. See Fig. 6 for an example.

The proof is by induction. The inductive step is the following: given $(\omega, T(l, h), t)$ such that $\bar{\omega}$ is not directed, the tower is parallel to the strip, all points of $T(l, h)$ lie on the W-side with respect to the base, and $\omega(0)$ does not belong to the line $x=x_1+t$, then there is a sequence of moves

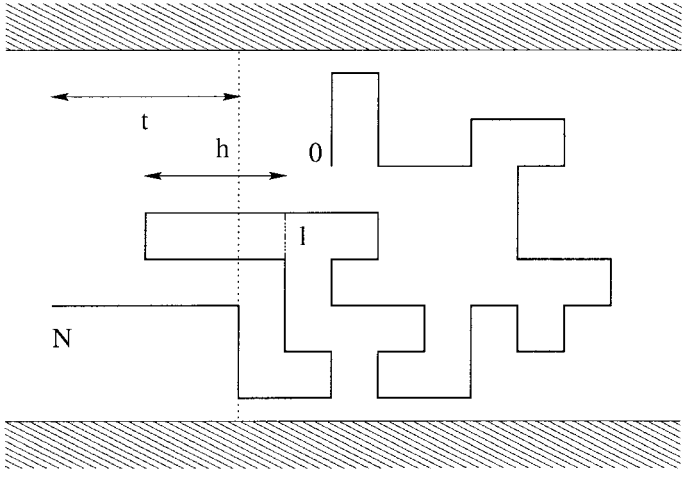

Fig. 6. Definition of t when $|x_1-x_N|=0$. The dotted line has equation $x=\bar{x}$, h is the height of the tower and l its base.

such that: (a) the new walk also has a tower parallel to the strip, lying on the W-side with respect to the base; (b) one of the following two possibilities is verified: (b1) t increases; (b2) t remains constant and h, the height of the tower, increases.

To start the induction, it is sufficient to assume $h=0$ and choose as l an arbitrary step perpendicular to the boundary of the strip (it exists since $\bar{\omega}$ is not a rod). At the end of the inductive process we will obtain a new walk, which we will continue to name ω, such that either $\bar{\omega}$ is directed or $\omega(0)$ belongs to the line $x=x_1+t$. In the first case, we have finished. The second case will be discussed at the end.

Let us now prove the inductive step. Let us first suppose that $x=x_1+t$ contains a C-turn $\bar{\omega}[k,m]$. Note that, because of the definition of t, $\bar{\omega}[k,m]$ will be of the form $WN^{m-k-i}E$ or $WS^{m-k-i}E$. Then, if the base of the tower does not yet belong to $\bar{\omega}[k+1,m-1]$, use B22 moves to move the tower so that the base is $\Delta\bar{\omega}(a)$, $k+1\leqslant a\leqslant m-2$, and the tower lies on the W-side of the base. It is easy to see that, thanks to the definition of t, this is always possible. If $\bar{\omega}[k,m]$ is unobstructed, then by applying L0 moves it is possible to modify $\bar{\omega}[k,m]$ in such a way that the subwalk of ω connecting $\bar{\omega}(a-1)$ and $\bar{\omega}(a+2)$ is a tower directed to W of height $h+1$. In this way, either t increases by one (requirement (b1)) or t remains constant but h increases by one (requirement (b2)).

If $\bar{\omega}[k,m]$ is obstructed, there exists an unobstructed C-turn $\bar{\omega}[i,j]$ (see Theorem 1). If the base of the tower does not belong to $\bar{\omega}[i,j]$, reduce it to a kink, cut it and increase the height of the tower by one. If it contains the base of the tower, let us notice that $\bar{\omega}[k,m]$ must have length at least 4 as it is obstructed. Therefore, there exists a step belonging to $\bar{\omega}[k+1,m-1]$ which does not belong to $\bar{\omega}[i,j]$. Then, move the tower on this link (this can be done simply by B22 moves if this link is not adjacent to the base of the tower, or by L0 moves if this is not the case) and at this point reduce $\bar{\omega}[i,j]$ to a kink, and then increase the height of the tower by one, using B22 moves. In both cases, thanks to the definition of t, the tower can always grow in the W-direction. In this case t remains constant, but h increases by one.

Let us now suppose that the line $x=x_1+t$ does not contain any C-turn. Since, by hypothesis, it does not contain $\omega(0)$, it can contain only a subwalk $\bar{\omega}[i,j]$ such that either $j=N-2h$ or $\bar{\omega}[j,N-2h]$ is a line perpendicular to R_1, lying on the W-side of $x=x_1+t$. If the base of the tower does not belong to $\bar{\omega}[i,j]$, with an appropriate choice of y-direction, we have $\omega[i,N]=S^{N-i-t}W^t$. Then, using L0 and L1 moves, it is trivial to modify the walk into a new one such that $\omega[i,N]=W^{N-i}$. Therefore, t increases by at least $|j-i|$, proving the inductive step. If the base of the tower belongs to $\bar{\omega}[i,j]$, with an appropriate choice of y-direction,

$\omega[i, N]$ is of the form $S^{k_1}W^hSE^hS^{k_2}W^t$ with $h>0$. Moreover, if x_l is the x-coordinate of $\omega(l)$, we have $x_l > \bar{x}$ for all $l<i$. We will now distinguish two cases: (1) $k_2>0$, (2) $k_2=0$.

If $k_2>0$, using L0 moves, we can rewrite it as $S^{k_1+k_2-1}W^hSE^hSW^t$. Then, using B22 moves followed by local transformations L0, we have

$$S^qW^hSE^hSW^t \to S^qW^{h+t}SE^hS \to S^qW^{h+t}S^2E^h \tag{9}$$

If $h>1$ we can repeatedly modify the walk as follows:

$$\begin{aligned} S^qW^{h+t}S^2E^h &\to S^qW^{h+t}S^2E^{h-1}N \to S^qW^{h+t}S^2ENE^{h-2} \\ &\to S^qW^{h+t+1}SE^h \to S^qW^{h+t+1}SE^{h-1}S \\ &\to S^qW^{h+t+1}S^2E^{h-1} \end{aligned} \tag{10}$$

In this way, we obtain a walk with the original form and $h=1$. But a walk $S^qW^pS^2E$ can be modified into $S^qW^pS^2W$, and then, by means of L0 and L1 moves, into a rod W^{p+q+3}. Therefore, in case (1), the subwalk $\omega[i, N]$ can be transformed into a rod.

Let us now consider the case $k_2=0$. Since $h>0$, we should have $t=0$ so that $\omega[i, N] \to S^{k_1}W^hSE^h$. If the endpoint of the walk does not belong to the boundary of the strip we can repeat the steps presented for case (1). If the endpoint belongs to the boundary and $k_1>0$, we can use L0 moves to modify the walk into $S^{k_1-1}W^hS^2E^h$, which can be transformed into a rod as discussed in case (1). If $k_1=0$, since $w \geqslant 2$, we can apply the following transformations: if $h=1$, then

$$\omega[i, N] = WSE \to WSW \to W^2S \to W^3 \tag{11}$$

if $h=2$, then

$$\begin{aligned} \omega[i, N] &= W^2SE^2 \to NWS^2E \to NWS^2W \\ &\to NW^2S^2 \to \cdots \to NW^4 \to \cdots \to W^5 \end{aligned} \tag{12}$$

if $h>2$, then

$$\begin{aligned} \omega[i, N] &= W^hSE^h \to NWSW^{h-2}SE^{h-1} \to NW^{h-1}S^2E^{h-1} \\ &\to NW^{h-1}S^2E^{h-2}N \to NW^{h-1}S^2ENE^{h-3} \\ &\to NW^hSE^{h-1} \to \cdots \to NW^hS^2E^{h-2} \to \cdots \to NW^{2h-3}S^2E \\ &\to NW^{2h-3}S^2W \to \cdots \to NW^{2h} \to \cdots \to W^{2h+1} \end{aligned} \tag{13}$$

Therefore, in all cases we deform $\omega[i, N]$ into a rod. For the new walk, the variable t increases at least by $2h+1$ as required.

The proof of the inductive step is complete. In this way, we have shown that in a finite number of steps we obtain a walk ω which is quotient-directed or is such that $x_0 = x_1 + t$. In the latter case, if $x_2 = x_0$, $\bar{\omega}$ is L-shaped and thus directed. Thus, we need only to study the case $x_2 > x_0$. Again we will proceed by induction. We assume that the walk ω has a tower parallel to the strip which lies on the E-side with respect to the base. At the beginning notice that R_2 must certainly contain a C-turn as it contains none of the endpoints. Then, move the tower oh this C-turn. This is always possible. Then, we show that, as long as $x_2 > x_0$ and the walk is not directed, we can modify it in such a way that the tower increases in height. The argument is exactly identical to the one we have previously discussed for the case in which a C-turn exists on the line $x = x_1 + t$. Therefore, in a finite number of steps, we obtain a walk with $x_2 = x_0$. As we already discussed, this walk is quotient-directed. ▌

It is now trivial to state the ergodicity theorem for the kink-kink bilocal algorithm which is a simple consequence of the lemmas we proved above:

Theorem 3. Consider a walk ω in a two-dimensional strip of width $w \geqslant 2$ and suppose that $\omega(0)$ is not a nearest neighbour of a boundary of the strip. Then, ω can be reduced to a given rod using the kink-kink bilocal algorithm.

The result we presented above applies only to the two-dimensional case. Indeed, the algorithm is not ergodic in three dimensions. For instance, consider the walk ($N = 18$)

$$\omega \equiv (-y)^2\,(x)(z)^2\,(-x)^2\,(-z)^3\,(x)^2\,(y)(z)^2\,(-x)(-y)^2 \qquad (14)$$

By direct enumeration, one can verify that it cannot be reduced to a rod.

The kink-kink bilocal algorithm can also be used to simulate ring polymers. It is enough to exclude the L1 moves. Such an algorithm was considered in refs. 38 and 39. It is clear that this algorithm is not ergodic in three dimensions since it does not change the knot type of the ring. We will now prove its ergodicity in a two-dimensional strip. We will need the following lemma:

Lemma 4. In two dimensions, consider a closed walk ω, with $N > 4$ steps. It contains an unobstructed C-turn $\omega[i, j]$ with $0 \leqslant i < j < N$.

Proof. Consider $w = \omega[0, N-1]$. It is not directed, and thus, by Theorem 1, it contains an unobstructed C-turn $\omega[i, j]$ with $0 \leqslant i < j < N$. Since $N > 4$, it is also unobstructed in ω. ▌

We can now prove the ergodicity of the algorithm.

Theorem 4. Consider a lattice point x in a two-dimensional strip and the ensemble $\mathscr{R}_{x,N}$ of N-step self-avoiding polygons such that $\omega(0) = \omega(N) = x$. The kink-kink bilocal algorithm without L1 moves is an ergodic algorithm for $\mathscr{R}_{x,N}$.

Proof. The proof is similar to that of Lemma 3. Consider a walk ω with tower $T(l, h)$ along the strip and let $\bar{\omega} = \omega/T(l, h)$.

We will now prove the following: if the length of $\bar{\omega}$ is larger than 4, we can modify the walk into a new one ω' with tower $T(l', h')$ parallel to the strip and $h' > h$. Therefore, in a finite number of steps, $\bar{\omega}$ is reduced to a square of length 4 and ω is a rectangle of height one. It is trivial to show that all these rectangles can be modified one into the other ending the proof.

To prove the previous statement, first notice that at least one of the boundaries of the enveloping rectangle $R[\bar{\omega}]$ perpendicular to the strip does not contain $\omega(0)$ and contains a C-turn $\bar{\omega}[k, m]$, $0 \leqslant k < m \leqslant N$. Using B22 moves, we first move the tower on a link $l \in \bar{\omega}[k+1, m-1]$, adding kinks on top of this link, outward with respect to $R[\bar{\omega}]$. If $\bar{\omega}[k, m]$ is unobstructed, we can increase the height of the tower using L0 moves; otherwise, by Lemma 4, there exists an unobstructed C-turn $\bar{\omega}[i, j]$ such that $\omega(0) \notin \bar{\omega}[i+1, j-1]$. As we already discussed in Lemma 3, such a C-turn can be reduced to a kink which is then moved on top of the tower, increasing its height. ∎

3.4. Ergodicity for the Extended Reptation Algorithm

We will now prove the ergodicity in two dimensions of the extended reptation algorithm. We do not know whether this algorithm is ergodic for larger values of d.

Theorem 5. The extended reptation algorithm is ergodic in $d = 2$, for $w \equiv w_2 \geqslant 1$.

Proof. The proof is similar to that of Lemma 3. We assume we have a walk ω with a tower $T(l, h)$ parallel to the x-axis. Then, we consider the enveloping rectangle $R[\bar{\omega}]$: R_1 and R_2 are the two sides of $R[\bar{\omega}]$ perpendicular to the boundary of the strip. We will now show that it is possible to deform the walk so that one of the endpoints belongs either to R_1 or to R_2 (defined for the new walk).

The proof is by induction: Consider a walk ω such that none of the endpoints belongs to R_1 or R_2, which has a tower $T(l, h)$ parallel to the

x-axis, l belonging to R_1. Then, we can deform it into a new walk with tower $T(l', h+1)$ parallel to the x-axis, l' belonging to R_1. To prove this statement, notice that R_1 must contain a C-turn $\bar{\omega}[i, j]$. If $l \notin \bar{\omega}[i+1, j-1]$, using B22 moves, we can modify the walk so that it has a tower of height h on a link belonging to $\omega[i+1, j-1]$ on the W-side with respect to R_1. Since the walk has a C-turn, by Theorem 1, it has an unobstructed C-turn. If $\bar{\omega}[i, j]$ is unobstructed, we can reduce it to a kink obtaining a tower of height $h+1$. If $\bar{\omega}[k, l]$ with $l < i$ or $k > j$ is unobstructed, we can reduce it to a kink and use a B22 move to move the kink on top of the tower increasing its height. We must, then consider the special cases in which $\bar{\omega}[i, j] \cap \bar{\omega}[k, l]$ is not empty and $\bar{\omega}[i, j]$ is obstructed. It is easy to realize that one should have $l = i+2$ or $k = j-2$ and that they can be treated similarly. If $l = i+2$, since $|j-i| \geqslant 4$ ($\bar{\omega}[i, j]$ is obstructed), we can modify the walk (if needed) so that the base of the tower does not belong to $\bar{\omega}[k, l]$. Then, we reduce it to a kink, and move it on top of the tower increasing its height. If $k = j-2$, we can proceed analogously. We have thus proved the inductive step. Therefore, in a finite number of steps we can modify the walk into a new one such that: (a) one of the endpoints belongs to R_1 or R_2; (b) the base of the tower belongs to R_1. If one of the endpoints belongs to R_2, using reptation moves in the E-direction, we can reduce the walk to a rod. If none of the endpoints belongs to R_2, an endpoint belongs to R_1 and R_2 contains a C-turn $\bar{\omega}[i, j]$. Then, by means of B22 moves, one can move the tower on one of the links belonging to $\bar{\omega}[i, j]$, so that all points of the tower lie on the E-side with respect to the base. Then, consider the endpoint belonging to R_1. Using reptation moves in the W-direction, we can reduce the walk to a rod. Finally note that, if $w \geqslant 1$, the two rods E^N and W^N can be deformed one into the other. We have thus proved the ergodicity of the algorithm. ▮

It is easy to see that the moves that are added to the reptation algorithm (L0 and B22) are necessary for the ergodicity of the algorithm. Indeed consider the walk with $N = 22$:

$$N^2W^2S^2ESE^2N^2E^2SES^2W^2N^2 \tag{15}$$

It is easy to see that it cannot be deformed without using L0 moves. The walk with $N = 13$

$$NWS^2E^2NE^2S^2WN \tag{16}$$

requires instead the B22 moves to be reduced to a rod.

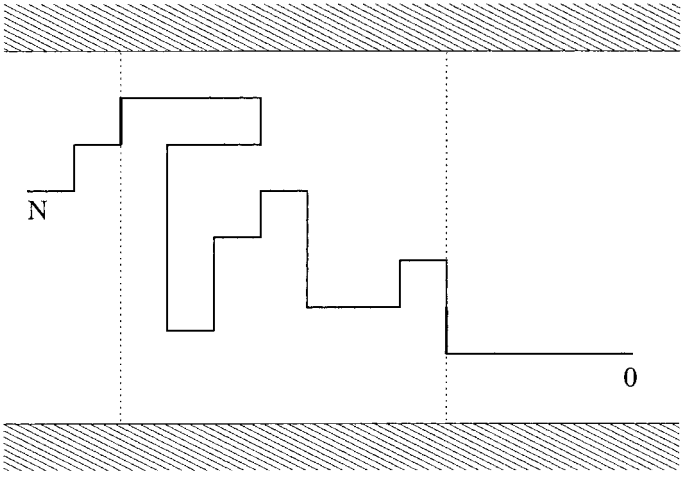

Fig. 7. With the definitions given in Theorem 6, the dotted vertical lines have equations $x = \bar{x}^{\pm}$. The step in boldface on the left is an S-step, while the step in boldface on the right is a step belonging to $x = \bar{x}^{+}$ that does not have an empty shadow and is not an S-step.

3.5. Ergodicity for the Kink-End Reptation Algorithm

In this Section we consider the kink-end reptation algorithm. We show that this algorithm is ergodic in any cylinder in $d \geqslant 3$ and in the presence of a single surface (or in free space) in two dimensions. This algorithm is not ergodic in a two-dimensional strip. However, ergodicity is recovered by adding L0 moves.

Theorem 6. For $d \geqslant 3$, the kink-end reptation algorithm is ergodic for $w \equiv \min_i w_i \geqslant 1$ and $N \geqslant 3$.

Proof. In order to prove the theorem, let us introduce some useful definitions. We say[12] that a walk step $\Delta\omega(j)$, $1 \leqslant j \leqslant N-4$, is an *S-step* (see Fig. 7) if $\Delta\omega(j-1) = \Delta\omega(j+1)$ are directed in the $(\pm x)$-direction, while $\Delta\omega(j)$ is orthogonal to them. Note that we indicate by x the first direction which, by definition, is always infinite. Given a walk step $\Delta\omega(j)$ which is not in the $(\pm x)$-direction, we also define its positive and negative *shadow*. If $\hat{x}$ is the unit vector $(1, 0, \ldots, 0)$, the positive shadow is the set of lattice points $\{\omega(j) + n\hat{x}, \omega(j+1) + n\hat{x} : n \in Z^{+}\}$, where Z^{+} is the set of positive integers. The negative shadow is defined analogously, considering $n \in Z^{-}$,

[12] It would be natural to let j be $N-3$ and $N-2$. However, in our proof the last two steps play a special role. By requiring $j \leqslant N-4$, we require that neither $\Delta\omega(j)$ nor $\Delta\omega(j+1)$ is one of the last two steps of the walk.

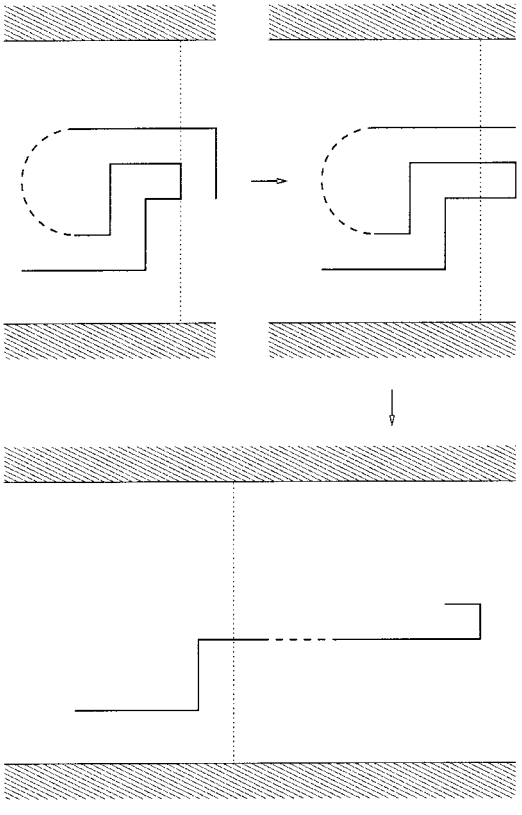

Fig. 8. The sequence of moves that generates a new walk ω' such that $\omega'[j, N-2]$ is a rod in the $(\pm x)$ direction. The dashed line in the upper part of the figure indicates an arbitrary subwalk. The dotted line is the line of equation $x = \bar{x}^+$.

Z^- being the set of negative integers. We say that $\Delta\omega(j)$ has an *empty shadow*, if its positive or negative shadow contains none of the walk sites $\omega(k)$, $0 \leqslant k \leqslant N-2$. To understand the relevance of this definition, note that, if $\Delta\omega(j)$ has an empty shadow, then (see Fig. 8) we can perform end-kink moves by adding kinks on top of $\Delta\omega(j)$, obtaining eventually a new walk ω' such that $\omega'[j, N-2]$ is a rod in the $(\pm x)$-direction. Note that, by definition, the last two walk sites may belong to the shadow. However, they do not represent an obstruction for the end-kink moves since the last two steps are deleted in the first iteration of the process.

The proof of the theorem is based on the following inductive step: given a walk ω such that the subwalk $\omega[i, N-2]$, $0 < i \leqslant N-2$, is a rod in the $(\pm x)$-direction, we can deform it into a new walk ω' such that, for

some $j<i$, $\omega'[j, N-2]$, is a rod in the $(\pm x)$-direction. Note that it is possible that, at the beginning of the induction process, $i=N-2$: it simply means that $\Delta\omega(N-3)$ is not in the $(\pm x)$-direction. This inductive step allows to prove that, in a finite number of steps, any walk can be deformed into $(\pm x)^{N-2} X$ where X is a two-step walk. Then, since $w \geqslant 1$ and $N \geqslant 3$, with an appropriate choice of the y-axis, we can deform it as follows:

$$(\pm x)^{N-2} X \to y(\pm x)(-y)(\pm x)^{N-2} \to (\pm x)^N \tag{17}$$

Thus all walks can be deformed into a rod $(\pm x)^N$. Finally, it is easy to show that the rod $(+x)^N$ can be deformed into $(-x)^N$, proving the ergodicity of the algorithm.

To prove the inductive step, let us introduce coordinates $\omega(k) = (x_k, y_k, z_k, \ldots)$ and define $\bar{x}^-$ (resp. $\bar{x}^+$) as the smallest (resp. largest) value of x such that there exists a walk site $\omega(k)$, $k<N-2$, with $x_k = \bar{x}^\pm$ and $\Delta\omega(k)$ not in the $(\pm x)$-direction. If $\omega[0, N-2]$ is not a rod in the $(\pm x)$-direction, $\bar{x}^-$ and $\bar{x}^+$ certainly exist although they may coincide.

Now, consider the links belonging to the hyper-surfaces $x=\bar{x}^-$ and $x=\bar{x}^+$. Suppose that one of them $\Delta\omega(l)$, $l<N-2$, is not an S-step. It is not restrictive to assume that it belongs to $x=\bar{x}^+$. We will now show that there exists $j<i$ such that the step $\Delta\omega(j)$ belongs to $x=\bar{x}^+$ and has an empty shadow. If $\Delta\omega(l)$ has an empty shadow, we can take $j=l$. Since $l<N-2$, and all steps $\Delta\omega(i), \ldots, \Delta\omega(N-3)$ are in the x-direction, we have $j<i$. If $\Delta\omega(l)$ does not have an empty shadow, there are two possibilities: (a) $\Delta\omega(l-1)$ is oriented in the negative x-direction; (b) $\Delta\omega(l+1)$ is oriented in the positive x-direction. We will consider only case (a) since case (b) is completely analogous. In case (a) (see Fig. 7) we will now show that $\Delta\omega(l+1)$ has an empty shadow and that $l+1<i$, so that we can take $j=l+1$. To prove this statement we will show the following: (a1) $l<N-3$; (a2) $\Delta\omega(l+1)$ belongs to $x=\bar{x}^+$; (a3) $\Delta\omega(l+2)$ cannot be oriented in the positive x-direction. From (a2) and (a3) we see that $\Delta\omega(l+1)$ has an empty positive shadow, while (a2) and (a1) allow to conclude $l+1<i$ as required. To prove (a1), note that, if (a1) were not true, we would have $l=N-3$ and the walk would be of the form $(-x)^{N-3} d_l X$, where $X \equiv \omega[N-2, N]$ and d_l is the direction of $\Delta\omega(l)$. But this implies that $\Delta\omega(l)$ has an empty (negative) shadow, which is against the initial assumption. To prove (a2), note that $\Delta\omega(l+1)$ cannot be oriented in the negative x-direction; otherwise, since $l<N-3$, $\Delta\omega(l)$ would be an S-step. If it were directed in the positive x-direction, then the walk would be $(-x)^l d_l x^{N-l-3} X$ where d_l is the direction of $\Delta\omega(l)$ and X indicates the last two steps. But in this case $\Delta\omega(l)$ would have an empty (negative) shadow. Therefore, (a2) is proved. If (a3) were not true, the walk would be $(-x)^l d_l d_{l+1} x^{N-l-4} X$,

where d_l and d_{l+1} are the directions of $\Delta\omega(l)$ and $\Delta\omega(l+1)$ respectively, and, therefore, $\Delta\omega(l)$ would have an empty (negative) shadow, against the initial hypothesis.

If $\Delta\omega(j)$ has an empty shadow, then we can deform the last steps of the walk as follows (we assume, without loss of generality, to have a positive empty shadow):

$$\omega[j, N] \to xy(-x)\cdots \to x^2y(-x)^2\cdots \to \cdots \to x^{N-2-j}y(-x) \qquad (18)$$

where y is the direction of $\Delta\omega(j)$ (see Fig. 8). We have thus proved the inductive step.

Let us now suppose that all walk steps belonging to $x=\bar{x}^-$ and $x=\bar{x}^+$ are S-steps. In this case it is easy to convince oneself that, with an appropriate choice of axes, the walk has the form x^pyXx^hY, where $p>0$, $h>0$, X is the subwalk $\omega[p+1, N-h-2]$ and Y is the configuration of the last two steps. Clearly, if $\omega(k)\in X$, then $\bar{x}^- < x_k \leqslant \bar{x}^+$. If $\bar{x}^- = \bar{x}^+$, X is empty and $\omega = x^pyx^hY$. Consider now $\Delta\omega(p)$ which is the only walk step[13] (as it can be seen from the explicit expressions above) belonging to the face $x=\bar{x}^-$. Since $w\geqslant 1$, it is possible to fix the positive z-direction in such a way that $\omega(p)+\hat{z}$ ($\hat{z}$ is the unit vector in the positive z-direction) is inside the cylinder. Then, we deform the walk as follows:

$$\omega \to zx(-z)\,x^{p-1}\cdots \to (-x)\,zx^2(-z)\,x^{p-1}\cdots \to (-x)^{N-2}\,zx \qquad (19)$$

Thus, we can take $j=0$, proving the inductive step. ∎

It is easy to see that this algorithm is not ergodic in a two-dimensional strip of width $w\equiv w_2$. Consider, for instance, the following walk of length $N=(4w+2)\,k+2$:

$$((EN)^w\,E(ES)^w\,E)^k\,E^2 \qquad (20)$$

$k\geqslant 1$. It is easy to verify that it cannot be modified by the algorithm.

The previous theorem can be extended to two dimensions in free space, or in the presence of a single boundary, i.e., for $w=\infty$.

Theorem 7. In two dimensions, the kink-end reptation algorithm is ergodic for $N\geqslant 3$, in the presence of a single boundary or in free space.

Proof. The proof is similar to that of Theorem 6. Again, we want to prove that, given a walk ω such that $\omega[i, N-2]$, $0<i\leqslant N-2$, is a rod in

[13] Note, however, that $\omega(N)$ may belong to the face $x=\bar{x}^-$. This is of no relevance for the following discussion since in an end-kink move the last two steps are deleted.

the $(\pm x)$-direction, we can deform it into a new one ω' such that, for some $j<i$, $\omega'[j, N-2]$ is a rod in the $(\pm x)$-direction. Using the same notations of Theorem 6, the reader can convince himself that we should only modify the proof of the inductive step in the case in which only S-steps belong to the lines $x=\bar{x}^-$ and $x=\bar{x}^+$. In this case, with a proper choice of axes, the walk has the form

$$\omega = E^p N X E^h Y \tag{21}$$

$p \geqslant 1$, $h \geqslant 1$, where X is the subwalk $\omega[p+1, N-h-2]$ which is contained between the lines $x=\bar{x}^{\pm}$ and $Y=\omega[N-2, N]$. If $\omega(p)$ does not belong to the boundary, we can modify the walk as follows:

$$\omega = E^p N X E^h Y \to SENE^{p-1} N X E^h \to WSE^2 NE^{p-1} \cdots \to W^{N-2} SE \tag{22}$$

so that we can take $j=0$, proving the inductive step.

We should finally discuss the case[14] in which $\omega[0, p]$ belongs to the boundary $y=0$. Let $\bar{y}$ be the largest value of n_y such that there exists a walk step $\Delta\omega(l)$, $l<N-2$, belonging to the line $y=n_y$. First we show that, if $y_k > \bar{y}$ (y_k is the y-coordinate of $\omega(k)$), then $k=N-1$ or $k=N$. Indeed, since $y_0=0$, if we had $y_k > \bar{y}$ and $k \leqslant N-2$, then $\omega[k-1, N-2]$ would point N which is in contrast with (21). Now, consider the sites belonging to the line $y=\bar{y}$ and let $\omega(k)$ be the site with smallest k. Clearly $k \leqslant i$. Note, moreover, that $\Delta\omega(k-1)$ is in the positive y direction. Indeed, it does not lie on $y=\bar{y}$, otherwise $\omega(k-1)$ would lie on this line (remember that $\omega(k)$ is the walk site with smallest k belonging to this line). It is not oriented in the negative y-direction, otherwise, $y_{k-1} > \bar{y}$. For the same reason, $\Delta\omega(k+1)$ is not in the positive y-direction, unless $k+1=N-2$. Then, we can use end-kink moves to put kinks on top of $\Delta\omega(k)$ in the positive y direction. There is no obstruction to these moves, since the only possible walk sites that can have a larger y are the last two sites that are removed in the first iteration of the process. In this way we modify the walk into a new one (we keep calling it ω) such that $\omega[k-1, N-2]$ is a rod directed in the positive y-direction.

Now, let $\tilde{x}$ be the largest x such that there exists a walk step $\Delta\omega(l)$, $l<N-2$, belonging to the line $x=\tilde{x}$. Consider the steps belonging to this line and let j be the smallest integer such that $\Delta\omega(j)$ belongs to $x=\tilde{x}$. Clearly $j \leqslant k-1 < k \leqslant i$. Since $\omega[0, p]$, $p>0$ is directed E and $\omega[k-1, N-2]$ is directed N, $\Delta\omega(j)$ has an empty positive shadow. Therefore, by

[14] If $p>1$, we can use the sequence of moves (22) adding kinks "above" the boundary. The difficult case corresponds to $p=1$.

means of end-kink moves, we modify the walk so that $\omega[j, N-2]$ is a rod in the positive direction. Since $j < i$, we have proved the inductive step. ∎

If one considers two dimensional strips, an ergodic algorithm can be obtained by adding L0 moves.

Theorem 8. In two dimensions the kink-end reptation algorithm with L0 moves is ergodic for $N \geqslant 3$ and $w_2 \geqslant 1$.

Proof. The proof is identical to that of Theorem 6. We should only change the proof of the inductive step for the case in which there are only S-steps on the lines $x = \bar{x}^{\pm}$. With a proper choice of axes the walk has the form (21). Then, using L0 moves, we modify the walk as follows

$$\omega = E^{p-1}NE \cdots \to \cdots \to NE^{p} \cdots \tag{23}$$

Then, using end-kink moves,

$$\omega \to WNE^{p+1} \cdots \to \cdots \to W^{N-2}NE \tag{24}$$

which can be deformed into a rod. ∎

Although the kink-end reptation algorithm is ergodic in two dimensions in the absence of confining surfaces, the proofs of the theorems indicate that "staircase" sections of the walk (for instance sections of the form $\cdots ENENENEN \cdots$) will be changed very slowly by the algorithm. Therefore, in order to have an efficient implementation, it is probably useful to include in all cases the L0 moves.

4. TRANSITION MATRICES

In the previous Section we have discussed the ergodicity of the algorithms. Now, we discuss how to implement them in order to obtain the correct probability distribution. Here we will discuss how to use them to generate walks with uniform probability in the ensembles $\mathscr{E}_{x,N}$ or $\mathscr{E}_N$. Any other probability distribution can be obtained by adding a Metropolis test or a generalization thereof.[15]

4.1. Kink-Kink Bilocal Algorithm

We will begin by considering the kink-kink bilocal algorithm. Although not necessary to ensure the ergodicity of the algorithm, we will

[15] For simple lattice models of homopolymers and proteins, the Metropolis criterion should also provide a plausible physical dynamics.[46] For a different point of view see refs. 47 and 48.

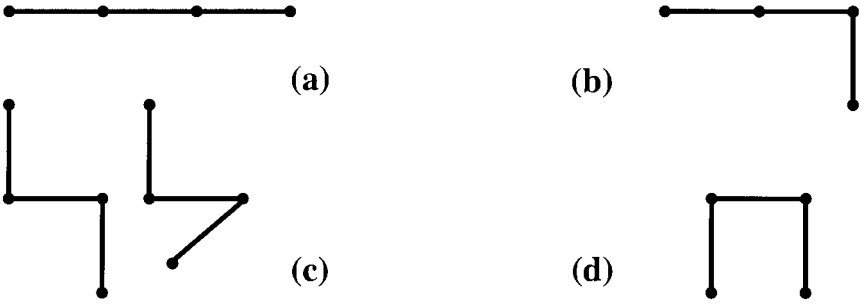

Fig. 9. Configurations of three consecutive links: (a) configuration of type I; (b) configuration of type L; (c) configuration of type S; (d) configuration of type U.

also add the local L00 (crankshaft) moves. In order to describe the algorithm it is important to classify the possible configurations of three successive links (see Fig. 9):

1. the bonds have the same direction (I configuration);

2. two consecutive bonds have the same direction, while the third one is perpendicular to them (L config.);

3. the first and the third bond are perpendicular to the second one, and they are either parallel or perpendicular to each other (S config.);

4. the first and the third bond are perpendicular to the second one, and they are antiparallel to each other (U config.).

The algorithm works as follows:

• Step 1. Choose a random site i of the current walk ω, $0 \leqslant i \leqslant N$. If $i = N$, propose an L1 move and go to step 5.

• Step 2. Determine the configuration of the subwalk $\omega[i-1, i+2]$. If $i = N-1$, we imagine adding a link $\Delta\omega(N)$ parallel to $\Delta\omega(N-1)$, so that the possible configurations are of type L and I. Analogously, if $i = 0$, we imagine adding a link $\Delta\omega(-1)$ parallel to $\Delta\omega(0)$.

• Step 3. Draw a random number r, uniformly distributed in $[0, 1]$. Depending on the configuration of $\omega[i-1, i+2]$, do the following:

1. I: If $r > (2d-2)\, p(22)$, perform a null transition and the iteration ends. Otherwise, go the next step.

2. L: If $r > (2d-3)\, p(22) + p(0)$, perform a null transition and the iteration ends. If $(2d-3)\, p(22) < r < (2d-3)\, p(22) + p(0)$, propose an L0 move and go to step 5. Otherwise, go to the next step.

3. S: If $r > (2d-4)\ p(22) + 2p(0)$ perform a null transition and the iteration ends. If $(2d-4)\ p(22) < r < (2d-4)\ p(22) + 2p(0)$ propose an L0 move: there are two possibilities which are chosen amongst randomly; then go to step 5. Otherwise, go to the next step.

4. U: If $r > 1 - (2d-3)\ p(00)$ propose an L00 move: there are $(2d-3)$ possibilities which are chosen amongst randomly; then, go to step 5. Otherwise, go to the next step.

• Step 4. Choose a second integer j uniformly in the disjoint intervals, $-1 \leqslant j \leqslant N$, $j \neq i-1, i, i+1$. If $j = -1$, N make a null transition and the iteration ends. Otherwise, depending on the configuration of $\omega[i-1, i+2]$, do the following:

— $\omega[i-1, i+2]$ is of type I, S, L: if $j=0$ or $j=N-1$, or if $\omega[j-1, j+2]$ is not of type U perform a null transition and the iteration ends. Otherwise, propose a B22 move, cutting the kink $\omega[j-1, j+2]$ and adding it to $\omega[i, i+1]$ in one of the possible directions.[16] Then, go to the next step.

— $\omega[i-1, i+2]$ is of type U: according to the configuration of $\omega[j-1, j+2]$ (if $j=0$, $N-1$ imagine adding links as before) do the following:

1. $\omega[j-1, j+2]$ is of type I: If $r < (2d-2)\ p(22)$ (note that the random number r appearing here is the same used in Step 3.), propose a B22 move: cut the kink $\omega[i-1, i+2]$ and add it on top of $\omega[j, j+1]$ in a possible random direction, and then go to step 5. Otherwise, perform a null transition and the iteration ends.

2. $\omega[j-1, j+2]$ is of type L: If $r < (2d-3)\ p(22)$, propose a B22 move: cut the kink $\omega[i-1, i+2]$ and add it on top of $\omega[j, j+1]$ in a possible random direction, and then go to step 5. Otherwise, perform a null transition and the iteration ends.

3. $\omega[j-1, j+2]$ is of type S: If $r < (2d-4)\ p(22)$ propose a B22 move: cut the kink $\omega[i-1, i+2]$ and add it on top of $\omega[j, j+1]$ in a possible random direction, and then go to step 5. Otherwise, perform a null transition and the iteration ends.

4. $\omega[j-1, j+2]$ is of type U: If $r < (2d-3)\ p(22)$, propose a B22 move: cut the kink $\omega[i-1, i+2]$ and add it on top of $\omega[j, j+1]$ in

[16] Note that "possible" directions means here a direction for which there are no intersections with $\omega[i-1, i+2]$. It is worth mentioning that an I (respectively S, L) always has $(2d-2)$ (respectively $(2d-4)$, $(2d-3)$) possible directions.

a possible random direction, and then go to step 5; if $(2d-3)\ p(22) < r < 2(2d-3)\ p(22)$, propose a B22 move: cut the kink $\omega[j-1, j+2]$ and add it on top of $\omega[i, i+1]$ in a possible random direction, and then go to step 5. Otherwise, perform a null transition and the iteration ends.

• Step 5. Check for self-avoidance. If the proposed new walk is self-avoiding keep it, otherwise perform a null transition.

The algorithm we have presented depends on three probabilities $p(0)$, $p(00)$ and $p(22)$ that are the probabilities of an L0, L00 and B22 move respectively. It is easy to check that the algorithm satisfies detailed balance so that the walks are generated with the correct probability distribution. We should now determine the single probabilities that must be such to satisfy the obvious constraint

$$\sum_{\omega'} P(\omega \to \omega') = 1 \tag{25}$$

Considering the configurations I, L, and S we obtain the constraints

$$(2d-2)\ p(22) \leqslant 1 \tag{26}$$

$$(2d-3)\ p(22) + p(0) \leqslant 1 \tag{27}$$

$$(2d-4)\ p(22) + 2p(0) \leqslant 1 \tag{28}$$

If $\omega[i-1, i+2]$ is of type U, we obtain, depending on the configuration of $\omega[j-1, j+1]$:

$$(2d-2)\ p(22) + (2d-3)\ p(00) \leqslant 1 \tag{29}$$

$$(2d-3)\ p(22) + (2d-3)\ p(00) \leqslant 1 \tag{30}$$

$$(2d-4)\ p(22) + (2d-3)\ p(00) \leqslant 1 \tag{31}$$

$$2(2d-3)\ p(22) + (2d-3)\ p(00) \leqslant 1 \tag{32}$$

These conditions impose for $d \geqslant 2$:

$$(4d-6)\ p(22) + (2d-3)\ p(00) \leqslant 1 \tag{33}$$

$$(2d-4)\ p(22) + 2p(0) \leqslant 1 \tag{34}$$

A solution of these inequalities which maximizes $p(0)$ and $p(00)$ at $p(22)$ fixed, is

$$p(0) = \frac{1}{2}\left[1 - (2d-4)\, p(22)\right] \tag{35}$$

$$p(00) = \frac{1}{2d-3}\left[1 - (4d-6)\, p(22)\right] \tag{36}$$

$$p(22) \leqslant \frac{1}{4d-6} \tag{37}$$

Since the L00 move is not necessary for the ergodicity of the algorithm, while the B22 one is essential to ensure a fast dynamics, it is natural to require $p(22)$ to be maximal, even if this implies $p(00) = 0$. Then, we obtain the following transition probabilities:

$$p(0) = \frac{d-1}{4d-6} \tag{38}$$

$$p(00) = 0 \tag{39}$$

$$p(22) = \frac{1}{4d-6} \tag{40}$$

In two dimensions $p(0) = p(22) = 1/2$, while in three dimensions $p(0) = 1/3$ and $p(22) = 1/6$.

4.2. Extended Reptation Algorithm

This algorithm extends the standard reptation method. The reptation (or slithering-snake) algorithm has two different implementations. The first one, which satisfies detailed balance, works as follows:

- Step 1. With probability 1/2 delete $\omega[N-1, N]$ and add a new link at the beginning of the walk; otherwise, delete $\omega[0, 1]$ and add a new link at the end of the walk.
- Step 2. Check if the new walk is self-avoiding. If it is keep it, otherwise perform a null transition.

A second version uses an additional flag which specifies which of $\omega(0)$ and $\omega(N)$ is the "active" endpoint. It works as follows:

- Step 1. Delete one bond at the "active" endpoint and append a new one at the opposite end of the walk.
- Step 2. If the new walk is self-avoiding keep it, otherwise stay with the old walk, and change the flag, switching the active endpoint.

This algorithm[17] does not satisfy detailed balance, but it satisfies the stationarity condition generating the correct probability distribution.

The extended reptation algorithm consists in combining with non-zero probability the reptation algorithm and the kink-kink bilocal algorithm. More precisely the algorithm works as follows:

- Step 1. With probability p perform a reptation move, with probability $1-p$ a kink-kink bilocal move, as specified in the previous section.

Note that in this algorithm the L1 moves are no longer needed. Therefore one can modify Step 1. of the kink-kink bilocal algorithm choosing i such that $0 \leqslant i \leqslant N-1$. The probability p is not fixed. It is only required that $0 < p < 1$ to ensure the ergodicity of the algorithm. It can therefore be tuned in order to obtain the best critical behaviour.

4.3. Kink-End Reptation Algorithm

The kink-end reptation algorithm uses kink-end and end-kink reptation moves (see Fig. 4). We will present here two different implementations of the algorithm which, however, are expected to have the same critical behaviour.

Let us explain the first implementation. An iteration consists of the following steps:

- Step 1. Choose a random site i of the current walk with $0 \leqslant i \leqslant N-2$.
- Step 2. Propose an end-kink move with probability $(2d-2)\, p(EK)$ or a kink-end move with probability $(2d-1)^2\, p(KE)$. In the first case delete the last two bonds of the walk and insert a kink on the bond $\Delta\omega(i)$ in one of the $(2d-2)$ possible orientations. In the second case, if $i \neq 0$ and $\omega[i-1, i+2]$ is a kink, remove it and attach two bonds at the end of the walk in one of the $(2d-1)^2$ possible ways. Otherwise, perform a null transition and the iteration ends.
- Step 3. Check if the proposed walk is self-avoiding. If it is keep it, otherwise make a null transition.

Detailed balance requires $p(KE) = p(EK) \equiv p$, while Eq. (25) implies

$$p \leqslant \frac{1}{(2d-1)^2 + (2d-2)} \tag{41}$$

[17] This implementation of the reptation algorithm has much in common with the hybrid algorithms used in simulations of lattice field theories, see footnote 42 in ref. 49.

Requiring p to be maximal, we obtain explicitly $p = 1/11$ in $d = 2$, and $p = 1/29$ in $d = 3$.

The second implementation of the algorithm is similar to that of the kink-kink bilocal algorithm. An iteration consists of the following steps:

- Step 1. Choose a random site of the current walk with $0 \leqslant i \leqslant N-2$.
- Step 2. Determine the configuration of the subwalk $\omega[i-1, i+2]$. If $i = 0$, we imagine adding a link $\Delta\omega(-1)$ parallel to $\Delta\omega(0)$.
- Step 3. Depending on the configuration of $\omega[i-1, i+2]$, do the following:

1. $\omega[i-1, i+2]$ is of type I, L, S: set $q = (2d-2)\, p(EK)$ if of type I, $q = (2d-3)\, p(EK)$ if of type L, $q = (2d-4)\, p(EK)$ if of type S. Then, with probability q propose an end-kink move, deleting the last two steps and adding a kink in one of the possible directions (see footnote 11). Otherwise, perform a null transition and the iteration ends.

2. $\omega[i-1, i+2]$ is of type U: with probability $(2d-3)\, p(EK)$ propose an end-kink move, deleting the last two steps and adding a kink in one of the possible $(2d-3)$ directions; with probability $(2d-1)^2\, p(KE)$ propose a kink-end move, cutting the kink and adding randomly two links to the walk in random directions. Otherwise, perform a null transition.

- Check whether the proposed new walk is self-avoiding. If it is keep it, otherwise make a null transition.

Detailed balance requires $p(KE) = p(EK) \equiv p$, while Eq. (25) gives the following constraints:

$$(2d-2)\, p(EK) \leqslant 1 \tag{42}$$

$$(2d-3)\, p(EK) \leqslant 1 \tag{43}$$

$$(2d-4)\, p(EK) \leqslant 1 \tag{44}$$

$$(2d-1)^2\, p(KE) + (2d-3)\, p(EK) \leqslant 1 \tag{45}$$

For $d \geqslant 1$, these inequalities give

$$p \leqslant \frac{1}{(2d-3) + (2d-1)^2} \tag{46}$$

The most efficient algorithm corresponds to taking the equality in Eq. (46). In two and three dimensions we obtain $p = 1/10$ $(d = 2)$ and $p = 1/28$ $(d = 3)$.

5. DYNAMIC CRITICAL BEHAVIOUR

In order to understand the efficiency of an algorithm one should analyze the autocorrelation time τ. There are several different definitions[18] for τ: the *exponential* autocorrelation time τ_{exp} that controls the relaxation of the slowest mode in the system and the *integrated* autocorrelation time $\tau_{\text{int},\mathcal{O}}$ that depends on the observable $\mathcal{O}$ one is considering and that controls the statistical errors on $\mathcal{O}$. For $N \to \infty$, one expects a dynamic critical behaviour, i.e., $\tau \sim N^z$, where the exponent z may depend on which autocorrelation time one is considering.

We now derive lower bounds on the exponent z in the absence of interactions. Let us consider global observables, like the squared end-to-end distance R_e^2 and the squared radius of gyration R_g^2. For bilocal algorithms we expect[10, 49] $\tau_{\text{int},\mathcal{O}} \gtrsim N^2$. The basic assumption is that the slowest mode appearing in global observables is associated to the relaxation of the squared radius of gyration R_g^2. Then, an estimate of τ can be obtained as follows. At each elementary step, R_g^2 changes by a quantity of order $N^{2\nu-1}$. An independent configuration is reached when the observable changes by one standard deviation $N^{2\nu}$. Assuming that the observable performs a random walk, we obtain $\tau \sim (N^{2\nu}/N^{2\nu-1})^2 \sim N^2$. In practice the argument should provide only a lower bound[19] which we expect to be correct for all global observables.

In particular, this should apply to the end-to-end distance R_e^2. Here, however, we should notice that our algorithms update the end-point of the walk with very different frequencies. The extended reptation and the kink-end reptation change $\omega(N)$ every $O(1)$ iterations, while the kink-kink bilocal algorithm updates $\omega(N)$ only every $O(1/N)$ iterations. Therefore, an additional factor of N should be added for the kink-kink bilocal algorithm: in this case, we expect $\tau \gtrsim N^3$. It is interesting to notice that the kink-kink bilocal algorithm behaves approximately as the algorithm of Reiter:[38] indeed, also in this algorithm, the endpoint is updated, with frequency $1/N$. The available numerical results for very short walks ($N \lesssim 100$) are in agreement with the bound given above: they indicate $\tau \sim N^3$ in two and three dimensions.

[18] We refer the reader interested in more precise and rigorous statements to refs. 11 and 49.

[19] In ref. 50 the following rigorous lower bound was obtained:

$$\tau_{\text{int},\mathcal{O}} \geqslant \frac{2\,\text{var}(\mathcal{O})}{C^2} - \frac{1}{2} \tag{47}$$

where $\text{var}(\mathcal{O})$ is the static variance of the observable $\mathcal{O}$ and C is the *maximum* change of $\mathcal{O}$ in a single Monte Carlo step. In the heuristic argument given above, we have replaced C by the *average* change of $\mathcal{O}$ in a single Monte Carlo step.

While Reiter's algorithm can be easily speeded up by increasing the frequency of the BEE and BKE moves, no improvement is possible for the kink-kink bilocal algorithm. Indeed, in this case it makes no sense to increase the frequency of the L1 moves. Clearly, endpoint moves should be performed with the same frequency of the moves that change the site $\omega(N-1)$, otherwise they do not effectively change the endpoint position. But $\omega(N-1)$ is updated by B22 and L0 moves with frequency $1/N$. Therefore, L1 moves should be performed with the same frequency.

Let us now discuss in more detail the different implementations of the extended reptation and of the kink-end reptation algorithm. For the extended reptation, we should choose between the two different implementations of the reptation dynamics. If one considers ordinary random walks, it is obvious that a new walk is generated in $O(N^2)$ iterations of the first algorithm and in exactly N iterations of the second one. Thus the second version is much more efficient than the first one. For SAWs we do not expect such a big difference since the walk will move in a given direction only for a small number of steps (≈ 8 in two dimensions, ≈ 14 in three dimensions in the absence of interactions). Therefore we expect an improvement by a constant factor, and, indeed, simulations[41] show that in three dimensions the second implementation is 5–6 times faster that the first one. In the extended reptation we should also fix the parameter p. From the discussion given above, it is clear that we must have $p > 0$ as $N \to \infty$, otherwise the motion of the endpoints slows down the dynamics. To fix its specific value, we may compare p to the probabilities of proposing local and bilocal moves in the kink-kink bilocal algorithm. Assuming that the probability of occuring of a I, L, U, and S configurations is independent of the position of the walk site—it should be approximately true for large values of N—the probability p_b of a bilocal move B22 and the probability p_l of a local move L0 are given by

$$p_b = 2p(22)\, p(\mathsf{U})[(2d-2)\, p(\mathsf{I}) + (2d-3)(p(\mathsf{L}) + p(\mathsf{U})) + (2d-4)\, p(\mathsf{S})] \tag{48}$$

$$p_l = p(0)(p(\mathsf{L}) + 2p(\mathsf{S})) \tag{49}$$

To have a quantitative prediction we should know $p(\mathsf{U})$, $p(\mathsf{I})$, $p(\mathsf{L})$, and $p(\mathsf{S})$. If we were considering non-reversal random walk we would have

$$p(\mathsf{U}) = \frac{2(d-1)}{(2d-1)^2} \tag{50}$$

$$p(\mathsf{I}) = \frac{1}{(2d-1)^2} \tag{51}$$

$$p(\mathsf{L}) = \frac{4(d-1)}{(2d-1)^2} \tag{52}$$

$$p(\mathsf{S}) = \frac{2(d-1)(2d-3)}{(2d-1)^2} \tag{53}$$

so that

$$p_b = \frac{16(d-1)^4}{(2d-3)(2d-1)^4} \tag{54}$$

$$p_l = \frac{4(d-1)^3}{(2d-3)(2d-1)^2} \tag{55}$$

In two and three dimensions we obtain $p_b \approx 0.1975$, $p_l \approx 0.4444$, and $p_b \approx 0.1365$, $p_l \approx 0.4267$ respectively. For SAWS the probabilities can be computed by means of a short Monte Carlo simulation. The results are reported in Table 1. In two dimensions we obtain $p_b = 0.096$, $p_l = 0.500$, while in three dimensions $p_b = 0.087$, $p_l = 0.446$. Thus, in the extended reptation algorithm, B22 moves are proposed with probability $\approx 0.1(1-p)$, while reptation moves are proposed with probability p. If one wants to balance these two types of moves—this is reasonable of one wants a physical kinetics—one should choose $p \sim 0.1$. On the other hand, it is, clear that reptation moves are more relevant than B22 moves. Indeed, reptation moves are essential for the motion of the endpoint, while B22 moves are required only to avoid the trapping of the endpoints. Thus we expect the algorithm to be more efficient for larger values of p. The simulations[41] indicate that the fastest dynamics is obtained for $0.5 \lesssim p \lesssim 0.9$.

Finally, let us consider the kink-end reptation algorithm. We presented two different versions and we discuss now their relative efficiency. The first implementation chooses the move without checking the nearby bonds. A deformation is always proposed but it may immediately fail because it does not respect self-avoidance when one considers the neighbours of the chosen bond. The second algorithm is more careful: the move is chosen after considering the position of the nearby bonds. However, with a finite

Table I. Probabilities of the Different Configurations of Three Links

d	$p(\mathsf{I})$	$p(\mathsf{L})$	$p(\mathsf{U})$	$p(\mathsf{S})$
2	0.152	0.481	0.108	0.259
3	0.051	0.356	0.102	0.491

probability, it performs a null transition. In order to compare correctly the two implementations we should therefore compute:

(a) for the first algorithm, the probability of bilocal moves that do not fail after checking the position of the two nearby bonds;

(b) for the second algorithm, the probability of proposing a bilocal move.

For both algorithms an easy computation gives

$$p_{\text{kink-end}} = p(\mathsf{U})\ p(2d-1)^2 \tag{56}$$

$$p_{\text{end-kink}} = p[(2d-2)\ p(\mathsf{I}) + (2d-3)(p(\mathsf{L}) + p(\mathsf{U})) + (2d-4)\ p(\mathsf{S})] \tag{57}$$

Therefore, the larger the value of p, the more efficient is the algorithm. Since p in the first implementation is smaller than in the second one, the second algorithm is more efficient than the first one. Of course, this should be expected since the second one chooses the proposed move more carefully. However, the improvement in efficiency is small, approximately 10% in two dimensions and only 3% in three dimensions.

ACKNOWLEDGMENTS

We thank Alan Sokal for many useful comments.

REFERENCES

1. P. G. de Gennes, *Scaling Concepts in Polymer Physics* (Cornell University Press, Ithaca, New York, 1979).
2. J. des Cloizeaux and G. Jannink, *Les polymères en solution* (Les Editions de Physique, Les Ulis, 1987); English translation: *Polymers in Solution: Their Modeling and Structure* (Oxford University Press, Oxford/New York, 1990).
3. K. F. Lau and K. A. Dill, *Macromolecules* **22**:3986 (1989).
4. A. Sali, E. Shakhnovich, and M. Karplus, *Nature* **369**:248 (1994).
5. P. H. Verdier and W. H. Stockmayer, *J. Chem. Phys.* **36**:227 (1962).
6. N. Madras and A. D. Sokal, *J. Stat. Phys.* **47**:573 (1987).
7. A. K. Kron, *Vysokomol. Soyed.* **7**:1228 (1965) [*Polymer Science USSR* **7**:1361 (1965)].
8. A. K. Kron, O. B. Ptitsyn, A. M. Skvortsov, and A. K. Fedorov, *Molek. Biol.* **1**:576 (1967) [*Molec. Biol.* **1**:487 (167)].
9. F. T. Wall and F. Mandel, *J. Chem. Phys.* **63**:4592 (1975).
10. F. Mandel, *J. Chem. Phys.* **70**:2984 (1979).
11. N. Madras and G. Slade, *The Self-Avoiding Walk* (Birkhäuser, Boston/Basel/Berlin, 1996).
12. K. Suzuki, *Bull. Chem. Soc. Japan* **41**:538 (1968).
13. S. Redner and P. J. Rednolds, *J. Phys. A* **14**:2679 (1981).
14. P. Grassberger, *Phys. Rev. E* **56**:3682 (1997).

15. M. Lal, *Molec. Phys.* **17**:57 (1969).
16. B. MacDonald, N. Jan, D. L. Hunter, and M. O. Steinitz, *J. Phys. A* **18**:2627 (1985).
17. N. Madras and A. D. Sokal, *J. Stat. Phys.* **50**:109 (1988).
18. N. Madras, A. Orlitsky, and L. A. Shepp, *J. Stat. Phys.* **58**:159 (1990).
19. S. Caracciolo, A. Pelissetto, and A. D. Sokal, *J. Stat. Phys.* **67**:65 (1992).
20. J. M. Deutsch, *J. Chem. Phys.* **106**:8849 (1997).
21. S. Caracciolo, A. Pelissetto, and A. D. Sokal, *Phys. Rev. Lett.* **72**:179 (1994), hep-lat/9307021.
22. M. Karplus and E. Shakhnovich, *Protein Folding* (Freeman, New York, 1992).
23. J. D. Bryngelson, J. N. Onuchic, N. D. Socci, and P. G. Wolynes, *Proteins: Struct. Funct. Genet.* **21**:167 (1995).
24. P. Grassberger, G. T. Barkema, and W. Nadler (eds.), *Monte Carlo Approach to Biopolymers and Protein Folding* (World Scientific, Singapore, 1998).
25. H. S. Chan and K. A. Dill, *J. Chem. Phys.* **99**:2116 (1993).
26. H. S. Chan and K. A. Dill, *J. Chem. Phys.* **100**:9238 (1994).
27. A. Sali, E. Shakhnovich, and M. Karplus, *J. Mol. Biol.* **235**:1614 (1994).
28. N. D. Socci and J. N. Onuchic, *J. Chem. Phys.* **101**:1519 (1994).
29. N. L. Nunes, K. Chen, and J. S. Hutchinson, *J. Phys. Chem.* **100**:10443 (1996).
30. A. Gutin, A. Sali, V. Abkevich, M. Karplus, and E. I. Shakhnovich, *J. Chem. Phys.* **108**:6466 (1998).
31. M. Cieplak, M. Henkel, J. Karbowski, and J. R. Banavar, *Phys. Rev. Lett.* **80**:3654 (1998).
32. O. Collet, *Monte Carlo Procedure for Protein Folding in Lattice Models: Conformational Rigidity*, cond-mat/9907191.
33. J. Skolnick and A. Kolinski, *J. Mol. Biol.* **221**:499 (1991).
34. A. Kolinski and J. Skolnick, *Proteins: Struct. Funct. Genet.* **18**:338 (1994).
35. D. G. Covell, *J. Mol. Biol.* **235**:1032 (1994).
36. P. E. Rouse, *J. Chem. Phys.* **21**:1272 (1953).
37. M. Doi, *Introduction to Polymer Physics* (Clarendon Press, Oxford, 1996).
38. J. Reiter, *Macromolecules* **23**:3811 (1990).
39. O. Jagodzinski, E. Eisenriegler, and K. Kremer, *J. Physique I* **2**:2243 (1992).
40. A. Kolinski, M. Milik, and J. Skolnick, *J. Chem. Phys.* **94**:3978 (1991).
41. S. Caracciolo, M. Papinutto, and A. Pelissetto, in preparation.
42. B. Berg and D. Foerster, *Phys. Lett. B* **106**:323 (1981).
43. C. Aragão de Carvalho and S. Caracciolo, *J. Phys. (France)* **44**:323 (1983).
44. C. Aragão de Carvalho, S. Caracciolo, and J. Fröhlich, *Nucl. Phys. B* **215**:209 (1983).
45. S. Caracciolo, A. Pelissetto, and A. D. Sokal, *J. Stat. Phys.* **63**:857 (1991).
46. H. Taketomi, Y. Ueda, and N. Go, *Int. J. Peptide Protein Res.* **7**:445 (1975).
47. R. Zwanzig, *Proc. Natl. Acad. Sci. USA* **94**:148 (1997).
48. H. S. Chan and K. A. Dill, *Proteins* **30**:2 (1998).
49. A. D. Sokal, Monte Carlo methods for the self-avoiding walk, in *Monte Carlo and Molecular Dynamics Simulations in Polymer Science*, K. Binder, ed. (Oxford University Press, Oxford/New York, 1995).
50. S. Caracciolo, A. Pelissetto and A. D. Sokal, *J. Stat. Phys.* **60**:1 (1990).