Efficient simulation of semiflexible polymers with stiff bonds

Barkema, G.T.; Leeuwen, J.M.J. van

Efficient simulation of semiflexible polymers with stiff bonds

Gerard T. Barkema[1,*] and J. M. J. van Leeuwen[2]
[1]*Department of Information and Computing Sciences, Universiteit Utrecht, Princetonplein 5, 3584 CC Utrecht, The Netherlands*
[2]*Instituut-Lorentz, Universiteit Leiden, Niels Bohrweg 2, 2333 CA Leiden, The Netherlands*
(Received 22 July 2016; published 6 January 2017)

We investigate the simulation of stiff (extensible) and rigid (inextensible) semiflexible polymers in solution. In particular, we focus on polymers represented as chains of beads, interconnected by bonds with a low to zero extensibility, and significant persistence in the bond orientations along the chain, whose dynamical behavior is described by the Langevin equation. We review the derivation of the pseudopotential needed for rigid bonds. The efficiency of a number of routines for such simulations is determined. We propose a routine for handling rigid bonds which is, for longer chains, substantially more efficient than the existing ones. We also show that for extensible polymers, the Rouse modes can be exploited to achieve highly efficient simulations. At realistic values for the extensibility, e.g., that of double-stranded DNA, the simulations are orders of magnitude faster than those for rigid bonds. With increasing stiffness, however, the allowable time step and hence the efficiency decreases, until a crossover point is reached below which the routines with rigid bonds are more efficient; we present a numerical estimate of this crossover point.

DOI: 10.1103/PhysRevE.95.012502

I. INTRODUCTION

The simulation of the dynamics of polymers is a key problem in polymer physics, with both fundamental aspects and great practical importance. As polymers are usually studied in solution, the thermal noise of the solvent is an important ingredient for the dynamics of the polymer chain. The influence of the solvent on the polymer is represented by a random force, which fluctuates rapidly with respect to the motion of the monomers of the polymer. Adding noise to the equation of motion turns the Newtonian equation into what is generically called the Langevin equation. The solution of the Langevin equation by a numerical simulation is a notoriously difficult problem for several reasons; to mention a few: the large number of dynamical variables, the proper handling of random noise, long-range interactions (hydrodynamical and self-avoidance), the wide separation of time scales and last but not least, the presence of strong binding forces, particularly if the forces between the monomers are so strong that the interdistance must be considered as rigid. Rigid distances act as constraints on the motion.

One reason to revisit the extensive literature on handling constraints concerns an unresolved issue in the derivation of the pseudopotential force that needs to be added to the equations of motion in order that the constraints remain satisfied. In physics one sees "rigid" as the limit of stiffer and stiffer bonds. This limit is singular as it implies that the number of degrees of freedom is reduced. The singularity of the limit is already manifest in the specific heat: as long as the N bonds are stiff, each of the $3(N+1)$ degrees of freedom carries $k_BT/2$, while for rigid bonds we have N constraints and only $2N+3$ degrees of freedom contribute to the specific heat. Thus handling the constraints by the introduction of generalized coordinates, adapted to the constraints, leads to a motion that differs from the behavior of very stiff bonds. The problem with the use of generalized coordinates has been discussed with the trimer as example. In the trimer with rigid bonds, the orientation of the two bonds, together with the irrelevant position of the center of mass, defines the configuration of the trimer. Already Kramers [1] showed that the solution of the Hamiltonian equations of motions in terms of generalized coordinates does not lead to the expected equilibrium distribution of the angle between the bonds. Subsequently Helfand [2] and van Kampen and Lodder [3] have analyzed this discrepancy and discussed the remedies.

For proper equations that describe motion of rigid bonds as the limit of stiff bonds, one has to acknowledge that the constrained subspace has a variable "thickness" as long as the bonds are stiff but not yet rigid. This may be seen as an entropy contribution to the energy, turning it into a free energy. The entropy is called a pseudopotential and its derivatives are the extra forces needed for the constrained subspace in order that the rigid bonds are the limit of stiff bonds. Fixman [4], as first, proposed to add this pseudopotential to the Hamiltonian. Later on a more complete analysis was given in the papers by Morse and co-workers [5,6].

Earlier, in a remarkable paper, Hinch [7] carefully analyzed the Brownian motion inside a constrained subspace and arrived also at the pseudopotential. In addition he found that another term had to be added, related to the dependence of the diffusivity on the configuration. This term was earlier proposed by Ermak and McCammon [8] in a different context. It is however disturbing that on the one hand the pseudopotential features to acknowledge the entropic wandering *around* the constraint (Morse), while on the other hand (Hinch) the pseudopotential is due to random motion *inside* the constrained subspace. Grassia *et al.* [9] elaborated on the analysis of Hinch and suggested that the variable diffusivity does not play a role provided one uses a higher order scheme for the numerical solution. We reanalyze the derivation of Hinch and conclude that the combination of the pseudopotential and the variable diffusivity does not contribute to the motion in the constrained subspace. The remaining "entropy" in the limit of stiff bonds is the true origin of the pseudopotential.

*Corresponding author: g.t.barkema@uu.nl

As Grassia *et al.* [9] point out, the use of a higher order scheme is necessary in order to make the solution "in principle exact," i.e., to guarantee that the solution approaches the exact solution in the limit of a vanishing time step. The reason is that first order schemes make spatial errors of order $(\delta x)^2$, which correspond to errors in time of order δt, for random forces. That implies that sizing down the time step does not help for the accuracy of the evolution in a finite time interval.

This brings us to the second reason to revisit the problem of rigid bonds. We show that one can do better than, e.g., the second order Runge-Kutta solution, by enforcing detailed balance on the solution scheme. The mathematical oriented literature, focusing on efficiency, tries to enlarge the order of convergence as function of the time step [10,11]. Rather than relying on general order considerations, we base our strategy on using physical quantities as criterion of convergence. In particular we exploit the condition that a simulation should obey detailed balance, which implies the proper equilibrium quantities. Of course the dynamical solution of the Langevin equations is not meant as a means to calculate the equilibrium properties, which can be obtained by simpler schemes such as Monte Carlo simulation or sometimes even by analytical means. But if a solution scheme does not provide the correct equilibrium properties, it is certainly incorrect. We show in addition that the bonus of detailed balance is also accurate time correlation functions.

The above considerations are particularly relevant for the dynamical behavior of the inextensible wormlike chain (IWLC), which is a widely used model for polymer behavior. In the IWLC the chain of monomers is replaced by a flexible continuum contour with fixed contour length. The dynamics of the model can be simulated by taking a fixed set of points at equal distances along the contour. The more points are taken on the contour, the better the behavior of the continuum chain is realized. In this way the IWLC is equivalent with a bead-spring model with rigid bonds, where the beads do not correspond to monomers. So in the dynamics of the IWLC one encounters the same difficulties as in a chain of beads connected by rigid bonds. It is a particular interesting question how efficient a solution scheme is for the dynamics of a bead-bond model for longer chains (or a more accurate simulation of the IWLC).

We start this paper by setting the context using a Hamiltonian for a bead-spring model, which on the one hand is versatile enough to describe extensible polymers, such as dsDNA and on the other hand is very well suited to study the limit of stiff to rigid bonds. Then we formulate an alternative solution of the Langevin equations in which detailed balance is enforced. This gives us the possibility to consider variations in the solution scheme and to calculate the optimal time step for the scheme that we propose. The next section introduces the pseudopotential for rigid bonds. We compare this with the expression given by Hinch [7] and show that this expression does not lead to a correction of the forces. This is a delicate point, since it is not evident from the form given by Hinch, but it follows from an equivalent formulation. This equivalent formulation is derived in Appendix B. With the machinery for rigid bonds available, we apply it to the trimer as an example, since the trimer is a historic playground for testing algorithms, because simulations are easy and fast. The trimer shows already the trend that a scheme obeying detailed balance is more accurate than, e.g., the second order Runge-Kutta method. This features becomes more pronounced for longer chains as we show in the next section. For completeness we discuss also what detailed balance implies for extensible chains, which can be effectively dealt with using modes. As a last section we present a calculation of the time-dependent correlation function of the squared deviation of the middle monomer, showing that the result is very accurate for the optimum time step of our solution scheme. The paper closes with a discussion of our results.

II. HAMILTONIAN CONTEXT

In this section we discuss the limit of stiff bonds to rigid bonds using a Hamiltonian consisting of a chain of beads held together by harmonic forces of the following simple form:

$$\mathcal{H} = \frac{\lambda}{2}\sum_{n=1}^{N}(|\mathbf{u}_n| - d)^2 - \kappa\sum_{n=1}^{N-1}\mathbf{u}_n\cdot\mathbf{u}_{n+1}, \tag{1}$$

which is still versatile enough to describe various kinds of polymers. $\mathbf{u}_n = \mathbf{r}_n - \mathbf{r}_{n-1}$ is the bond vector between the beads at successive positions $\mathbf{r}_{n-1}$ and $\mathbf{r}_n$. The beads are not necessarily associated with monomers. In a coarse-grained description the beads stand for groups of monomers. The parameters λ and κ represent the bond stiffness and the bending flexibility of the chain and the parameter d provides a length scale for the distance between monomers. With these adaptable parameters one can fit the force-extension curve of Wang *et al.* [12], which has been shown to well describe the measured force-extension curve of several semiflexible polymers [13]. The persistence length l_p is mostly determined by the parameter κ. In general the parameter λ is large with respect to κ. Their ratio will be denoted by ν,

$$\nu = \kappa/\lambda. \tag{2}$$

In this respect, double-stranded DNA, for which $\nu = 0.35$ is found [14], is only moderately stiff. The limit $\nu \to 0$, at fixed κ, turns the bonds from "stiff" to "rigid." The polymer f-actin is an example which approaches the rigid limit. The main theme of this paper is to study the limit of $\nu \to 0$.

Since the monomers are small in size, the viscous terms dominate over the acceleration terms. This leads to the Langevin equation of the form

$$\frac{d\mathbf{r}_n}{dt} = -\frac{1}{\xi}\frac{\partial\mathcal{H}}{\partial\mathbf{r}_n} + \mathbf{g}_n. \tag{3}$$

ξ is the friction coefficient and $\mathbf{g}_n$ is the thermal random force. The fluctuation-dissipation theorem requires the correlation function between the random forces to be given by

$$\left\langle g_m^\alpha(t) g_n^\beta(t')\right\rangle = \frac{2k_bT}{\xi}\,\delta^{\alpha,\beta}\,\delta_{m,n}\,\delta(t-t'). \tag{4}$$

In order not to be confused by irrelevant constants, we express distances in terms of d and scale the time t to a dimensionless variable τ,

$$t = c\tau, \tag{5}$$

where c is a time unit. c will be chosen such that amplitude of the correlation between the random forces equals 2. The

equation of motion then obtains the form

$$\frac{d\mathbf{r}'_n}{d\tau} = -\frac{\partial \mathcal{H}'}{\partial \mathbf{r}'_n} + \mathbf{g}'_n. \tag{6}$$

The scaled random force is given by

$$\mathbf{g}'_n = \frac{c}{d}\mathbf{g}_n. \tag{7}$$

As the strength of its correlations should become equal to 2 we have the equation

$$2\frac{c}{d^2}\frac{k_B T}{\xi} = 2, \tag{8}$$

which implies for c

$$c = \frac{\xi d^2}{k_b T} = \frac{1}{b^2}\frac{a^2 \xi}{k_b T}, \tag{9}$$

where we have converted d to the physical interbead distance $a = d/(1-2\nu) \simeq 0.3$ nm. For standard circumstances $c \simeq 52$ ps. This time scaling differs from the scaling that we used for extensible chains [15] and it corresponds to the scaling used in simulations of the IWLC [16,17].

The scaled Hamiltonian contains instead of λ and κ the dimensionless parameters

$$\Lambda = \frac{c\lambda}{\xi} \quad \text{and} \quad J = \frac{c\kappa}{\xi}. \tag{10}$$

J is the persistence length l_p in units of a bond length a. The scaled Hamiltonian then reads explicitly

$$\mathcal{H}' = \frac{\Lambda}{2}\sum_{n=1}^{N}(|\mathbf{u}'_n| - 1)^2 - J\sum_{n=1}^{N-1}\mathbf{u}'_n \cdot \mathbf{u}'_{n+1}. \tag{11}$$

We have chosen the symbol J because of the analogy with a system of coupled classical spins $\mathbf{u}_n$, where the nearest neighbor interaction is usually denoted by the symbol J.

From now on we drop the primes on the quantities. With forces due to the interaction via the Hamiltonian $\mathcal{H}$ represented by $\mathbf{f}_n$, the Langevin Eq. (5) gets the form

$$\frac{d\mathbf{r}_n}{d\tau} = \mathbf{f}_n + \mathbf{g}_n. \tag{12}$$

In the case of rigid bonds we have to obey the constraints

$$u_n = |\mathbf{u}_n| = 1, \quad \text{with} \quad \mathbf{u}_n = \mathbf{r}_n - \mathbf{r}_{n-1}. \tag{13}$$

which require additional tension forces in Eq. (12).

As the literature has few estimates [16] for the appropriate time steps in a reliable simulation, we specifically address this issue. A simulation is correct if it produces the correct time-dependent correlation functions. We have found that the most delicate correlation function is that of the mean squared displacement (MSD) of the middle monomer, since it is most sensitive to the slowest modes in the system and therefore exhibits the largest fluctuations. It is given by the expression

$$C(\tau) = \langle|\mathbf{r}^{\rm mid}(\tau) - \mathbf{r}^{\rm mid}(0)|^2\rangle, \tag{14}$$

where $\mathbf{r}^{\rm mid}(\tau)$ is the time-dependent position of the middle monomer and the average is over an equilibrium ensemble. In order to verify that the temporal behavior is properly reproduced in a simulation, one has to simulate substantially longer than the longest correlation time of the system, such that fluctuations on that time scale are sufficiently averaged out. As the correlation between the initial position and that on later times vanishes in the long run, the asymptotic behavior is given by

$$C(\infty) = \langle[|\mathbf{r}^{\rm mid}(\infty)|^2 + |\mathbf{r}^{\rm mid}(0)|^2]\rangle = 2\langle|\mathbf{r}^{\rm mid}(0)|^2\rangle. \tag{15}$$

So the correlation function should saturate to twice the equilibrium value. For rigid bonds one has an explicit expression for the equilibrium $C(\infty)$ [13].

III. EQUIVALENT FORMULATION

For the numerical solution of Eq. (12) with a time step $\Delta\tau$, one first generates random forces $\bar{\mathbf{g}}_n$ with the Gaussian probability

$$\mathcal{P}(\bar{\mathbf{g}}_n) \sim \exp\left(-\frac{|\bar{\mathbf{g}}_n|^2}{2}\right). \tag{16}$$

Then an increment $\Delta\mathbf{r}_n$ is calculated as

$$\Delta\mathbf{r}_n = \mathbf{f}_n\Delta\tau + \sqrt{2\Delta\tau}\,\bar{\mathbf{g}}_n. \tag{17}$$

We assume that the forces $\mathbf{f}_n$ are evaluated at midpoint,

$$\mathbf{r}_n^{\rm mid} = \mathbf{r}_n + \tfrac{1}{2}\Delta\mathbf{r}_n. \tag{18}$$

This turns the algorithm into a higher order scheme, but presents a computational problem since the midpoint depends on the increment to be calculated. The determination of the midpoint is an important ingredient in our analysis. Solving Eq. (17) for $\bar{\mathbf{g}}_n$, the probability on a displacement $\Delta\mathbf{r}_n$ becomes

$$\mathcal{P}(\Delta\mathbf{r}_n) \sim \exp\left(-\frac{|\Delta\mathbf{r}_n - \mathbf{f}_n\Delta\tau|^2}{4\Delta\tau}\right). \tag{19}$$

The solution of the Langevin equation through finite time steps as in Eq. (17) is equivalent with sampling the distribution (19). It is a Gaussian around the position shifted by the systematic force during $\Delta\tau$.

We may rephrase the probability by working out the exponent to order $\Delta\tau$,

$$\frac{|\Delta\mathbf{r}_n - \mathbf{f}_n\Delta\tau|^2}{4\Delta\tau} = \frac{|\Delta\mathbf{r}_n|^2}{4\Delta\tau} - \frac{1}{2}\mathbf{f}_n \cdot \Delta\mathbf{r}_n + \mathcal{O}(\Delta\tau). \tag{20}$$

The sum over the forces can be rewritten as

$$\sum_n \mathbf{f}_n \cdot \Delta\mathbf{r}_n \simeq \mathcal{H}(\mathbf{r}_n) - \mathcal{H}(\mathbf{r}_n + \Delta\mathbf{r}_n) + \mathcal{O}(\Delta\mathbf{r}_n)^3. \tag{21}$$

Using approximation (20) enables us to rewrite the probability on $\Delta\mathbf{r}_n$ as

$$\mathcal{P}(\Delta\mathbf{r}_n) \sim \exp\left(-\sum_n \frac{|\Delta\mathbf{r}_n|^2}{4\Delta\tau} + \frac{1}{2}[\mathcal{H}(\mathbf{r}_n) - \mathcal{H}(\mathbf{r}_n + \Delta\mathbf{r}_n)]\right). \tag{22}$$

Sampling the distribution (22) is equivalent to solving the Langevin equation. It looks a bit awkward since the probability involves the value of the energy at the configuration which has as yet to be chosen, but, as we shall see, this is surmountable. The technical implementation of the sampling of the distribution (22) will be discussed below.

Equation (22) for the probability has a number of advantages:

(i) The expression is symmetric with respect to forward and backward motion. This guarantees that the energy cannot grow.

(ii) The probability obeys detailed balance as can be seen as follows. Call P^+ the forward probability from configuration $\mathbf{r}_n$ to $\mathbf{r}_n + \Delta\mathbf{r}_n$ and P^- the probability for a move in the opposite direction. Then

$$\frac{P^+}{P^-} = \exp\left[\mathcal{H}(\mathbf{r}_n) - \mathcal{H}(\mathbf{r}_n + \Delta\mathbf{r}_n)\right], \tag{23}$$

which is indeed the detailed balance condition. Note that detailed balance even holds if we were to include the $\mathcal{O}(\Delta\tau)$ term, which is omitted in Eq. (20). Detailed balance guarantees that the motion produces correct equilibrium averages.

(iii) Since the probability (22) involves only energies one could refrain from calculating forces, which often is more cumbersome.

(iv) One can sample this distribution in several ways. E.g., one may first generate a new configuration with a diffusive probability

$$\mathcal{P}_1(\Delta\mathbf{r}_n) \sim \exp - \sum_n \left(\frac{|\Delta\mathbf{r}_n|^2}{4\Delta\tau} \right) \tag{24}$$

and realize the second factor

$$\mathcal{P}_2(\Delta\mathbf{r}_n) \sim \exp \tfrac{1}{2}[\mathcal{H}(\mathbf{r}_n) - \mathcal{H}(\mathbf{r}_n + \Delta\mathbf{r}_n)] \tag{25}$$

via an accept-or-reject procedure, which, as mentioned, obeys detailed balance.

(v) In certain cases one can conveniently split the Hamiltonian into an easy part and a correction,

$$\mathcal{H} = \mathcal{H}_0 + \Delta\mathcal{H}, \tag{26}$$

where the easy part $\mathcal{H}_0$ allows an explicit motion and the difficult part $\Delta\mathcal{H}$ can be handled approximately. This option is relevant for the discussion of stiff bonds (Sec. VII).

Implementation of the sampling

We face the challenge to sample a normalized distribution $P(x)$ which cannot be sampled easily. A well-known technique to overcome this difficulty [18] is to sample first a different (normalized) distribution $P'(x)$ which is an approximation of $P(x)$ and which can be sampled easily (for instance a Gaussian distribution). Samples drawn from $P'(x)$ are then accepted with probability $P(x)/[RP'(x)]$, where R is chosen large enough such that $P(x)/[RP'(x)] < 1$ for all x. The average acceptance probability is by construction equal to $1/R$.

The probability $\mathcal{P}_2(\Delta\mathbf{r}_n)$ can only be realized successfully by an accept-or-reject procedure if the exponent is close to zero, rendering the values of $\mathcal{P}_2$ to a small range around 1. For sufficiently small time steps indeed the exponent becomes small. If it is not small the range of the probabilities becomes large and one has to generate a random number in a large range R in order to compare the values of $\mathcal{P}_2$ with the random number for deciding whether to accept or not. But a large range of random numbers gives a low fraction of acceptance, roughly as $1/R$. That slows down the simulation by the same rate, as only the successful moves count in the time average. For any finite R we will encounter rare cases where the acceptance probability exceeds 100%, which is not feasible and hence results in undersampling. If these cases are very rare (and hardly observed in a normal simulation), detailed balance is practically obeyed.

One can improve the sampling by changing both $\mathcal{P}_1$ and $\mathcal{P}_2$. Instead of sampling the distribution Eq. (24) for $\mathcal{P}_1$ one may sample

$$\mathcal{P}_1'(\Delta\mathbf{r}_n) \sim \exp - \sum_n \left(\frac{|\Delta\mathbf{r}_n|^2}{4\Delta\tau} - \frac{1}{2}\mathbf{f}_n \cdot \Delta\mathbf{r}_n \right) \tag{27}$$

and correspondingly instead of Eq. (25) for $\mathcal{P}_2$ the distribution

$$\mathcal{P}_2'(\Delta\mathbf{r}_n) \sim \exp \frac{1}{2}\left[\mathcal{H}(\mathbf{r}_n) - \sum_n \mathbf{f}_n \cdot \Delta\mathbf{r}_n - \mathcal{H}(\mathbf{r}_n + \Delta\mathbf{r}_n) \right]. \tag{28}$$

Now the exponent of Eq. (28) vanishes to a higher order, if the forces are evaluated at the midpoint. We have insisted on the evaluation of the forces in the midpoint, as it turns not only the algorithm into a higher order scheme (which is essential for proper dealing of the random forces), but it also enables us to guarantee detailed balance. We will see that finding the midpoint is not without extra effort, but it pays off, as it allows us to make larger time steps and more accurate simulations.

IV. DYNAMICS IN A CONSTRAINED SUBSPACE

In case of constraints like Eq. (13) one has to introduce tensions that prevent the beads from violating the constraints. Such tensions are forces in the direction perpendicular to the constraint. The strengths are Lagrange multipliers T_a where the index a refers to the constraint $u_a = 1$. So the dynamic equation for the constrained subspace becomes

$$\dot{\mathbf{r}}_n = \mathbf{f}_n^{\text{unc}} + \sum_a T_a \, \mathbf{u}_n^a, \tag{29}$$

where $\mathbf{u}_n^a$ is the gradient of the bond u_a

$$\mathbf{u}_n^a = \frac{\partial u_a}{\partial \mathbf{r}_n}. \tag{30}$$

The force $\mathbf{f}_n^{\text{unc}}$ is the unconstrained force, the sum of the random and systematic forces acting on bead n. The values of the tensions follow from the velocity constraint

$$0 = \dot{u}_a = \sum_n \mathbf{u}_n^a \cdot \dot{\mathbf{r}}_n = \sum_n \mathbf{u}_n^a \cdot \left(\mathbf{f}_n^{\text{unc}} + \sum_b T_b \, \mathbf{u}_n^b \right). \tag{31}$$

The solution of the tension employs the inverse of the matrix [19]

$$G_{a,b} = \sum_n \mathbf{u}_n^a \cdot \mathbf{u}_n^b. \tag{32}$$

With the "geometrical" projector $\mathbf{P}_{n,m}$,

$$\mathbf{P}_{n,m} = \mathbf{I}\,\delta_{n,m} - \sum_{a,b} \mathbf{u}_n^a \, G_{a,b}^{-1} \, \mathbf{u}_m^b, \tag{33}$$

we can eliminate the tension and write the dynamic equation as

$$\dot{\mathbf{r}}_n = \sum_m \mathbf{P}_{n,m} \cdot \mathbf{f}_m^{\rm unc}. \tag{34}$$

Apart from projecting the motion to the constrained subspace one has to include the entropic effect of the width of the constraints in the limit of very stiff bonds. In Appendix A we find for this entropy

$$\mathcal{S} = -\tfrac{1}{2}\ln(\det G). \tag{35}$$

The derivatives of S with respect to the coordinates give additional forces on the beads.

By considering higher order effects in the constraint forces, Hinch [7], however, argued that in addition one has two extra forces,

$$\langle \Delta\dot{\mathbf{r}}_n \rangle = \sum_m \left(\mathbf{P}_{n,m} \cdot \frac{\partial}{\partial \mathbf{r}_n}\mathcal{S} + \frac{\partial}{\partial \mathbf{r}_m} \cdot \mathbf{D}_{n,m} \right). \tag{36}$$

The first term is the projection of the entropic force. The second term in Eq. (36) is due to the variation in the diffusivity

$$\mathbf{D}_{n,m} = \langle \dot{\mathbf{r}}_n \, \mathbf{r}_m \rangle. \tag{37}$$

The derivation of Hinch for the general case of different masses and friction coefficients is quite involved. Therefore we present in Appendix B the derivation which takes advantage of the simplification due to equal masses and friction coefficients. For instance one of the simplifications is that the diffusivity equals the projector,

$$\mathbf{D}_{n,m} = \mathbf{P}_{n,m}. \tag{38}$$

Our derivation leads to the expression

$$\langle \Delta\dot{\mathbf{r}}_n \rangle = -\sum_{k,m,a,b} \mathbf{u}_n^a \cdot G^{-1}_{a,b} \, \mathbf{u}_{km}^b \, \mathbf{D}_{k,m}, \tag{39}$$

We also show in the same Appendix that this expression is equivalent with the expression (36). We have to project the forces on to the subspace with the projector $\mathbf{P}$. Applying the projector to the expression (39) one sees that it vanishes since

$$\sum_n \mathbf{P}_{m,n} u_n^a = 0. \tag{40}$$

So there is no contribution of the sum of the two terms in Eq. (36) in the constrained subspace!

V. TRIMERS WITH RIGID BONDS

As we mentioned, the problem with random noise is that a first order scheme for solving the Langevin equations contains errors, which do not disappear for small time steps. In this section we illustrate this point taking the trimer as an example, which is an ideal test ground, since it contains all the aspects of longer fragments of polymers, while it permits analytic calculation of the various forces.

The trimer has three beads and two bonds $\mathbf{u}_1$ and $\mathbf{u}_2$, which are the only relevant dynamical variables, as we are not interested in the trivial center-of-mass motion. The matrix G as defined in Eq. (32) is a 2×2 matrix,

$$G = \begin{pmatrix} 2 & -C \\ -C & 2 \end{pmatrix} \quad \text{with} \quad C = \hat{\mathbf{u}}_1 \cdot \hat{\mathbf{u}}_2, \tag{41}$$

leading to the entropy $\mathcal{S}$,

$$\mathcal{S} = \frac{1}{\sqrt{4 - C^2}}. \tag{42}$$

So it is not difficult to obtain explicit equations for the motion of the bonds of the trimer. In this section we set the bending modulus $J = 0$, since we want to see the influence of the random forces on the distribution of the angles between the bonds. For $J = 0$ the distribution is a constant. A finite value of J gives a bias, which overshadows the subtleties of the random forces.

We give the equations for the update of the bonds which are derived from those for the beads as

$$\dot{\mathbf{u}}_a = \dot{\mathbf{r}}_a - \dot{\mathbf{r}}_{a-1}. \tag{43}$$

The bonds therefore feel the random forces

$$\mathbf{h}_a = \mathbf{g}_a - \mathbf{g}_{a-1}. \tag{44}$$

We have to generate first the independently Gaussian distributed $\mathbf{g}_n$ and then compute from them the $\mathbf{h}_a$. In addition to the random force we have the entropic force reading

$$\begin{aligned} \mathbf{s_1} &= [(\mathbf{2} + \mathbf{C})\hat{\mathbf{u}}_\mathbf{2} - (\mathbf{1} + \mathbf{2C})\hat{\mathbf{u}}_\mathbf{1}]/(\mathbf{4} - \mathbf{C^2}), \\ \mathbf{s}_2 &= [(2 + C)\hat{\mathbf{u}}_1 - (1 + 2C)\hat{\mathbf{u}}_2]/(4 - C^2). \end{aligned} \tag{45}$$

So the total, unprojected, force equals, by the absence of other systematic forces, $\mathbf{f}_a = \mathbf{h}_a + \mathbf{s}_a$.

As we have only two bonds we can easily solve for the tensions. This results in the explicit equations

$$\begin{aligned} \dot{\mathbf{u}}_1 &= \mathbf{f}_1 + [(C\hat{\mathbf{u}}_2 - 4\hat{\mathbf{u}}_1)(\hat{\mathbf{u}}_1 \cdot \mathbf{f}_1) \\ &\quad + 2(\hat{\mathbf{u}}_2 - C\hat{\mathbf{u}}_1)(\hat{\mathbf{u}}_1 \cdot \mathbf{f}_2)]/(4 - C^2), \\ \dot{\mathbf{u}}_2 &= \mathbf{f}_2 + [(C\hat{\mathbf{u}}_1 - 4\hat{\mathbf{u}}_2)(\hat{\mathbf{u}}_2 \cdot \mathbf{f}_2) \\ &\quad + 2(\hat{\mathbf{u}}_1 - C\hat{\mathbf{u}}_2)(\hat{\mathbf{u}}_2 \cdot \mathbf{f}_1)]/(4 - C^2). \end{aligned} \tag{46}$$

Note that the equations fulfill the constraints

$$\dot{\mathbf{u}}_a \cdot \hat{\mathbf{u}}_a = 0, \quad a = 1,2. \tag{47}$$

In Fig. 1 we have plotted two simulations of Eqs. (46) in which we determine the midpoint by iteration. The start (0 iterations) is a forward Euler scheme in which the forces and the tensions are computed in the starting point $\mathbf{u}_a$. The case

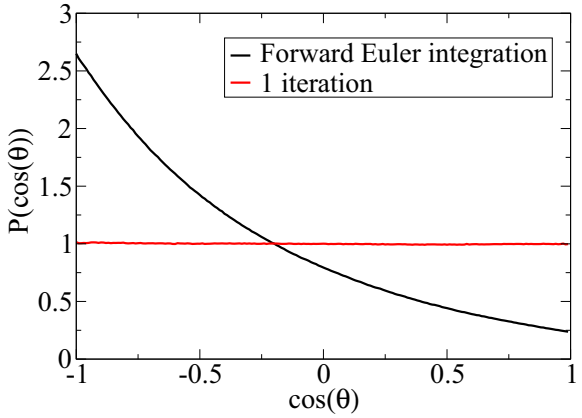

FIG. 1. Distribution of $\cos(\theta)$ for the trimer. The forward Euler curve corresponds to no iterations. The first iteration equals the second order Runge-Kutta routine.

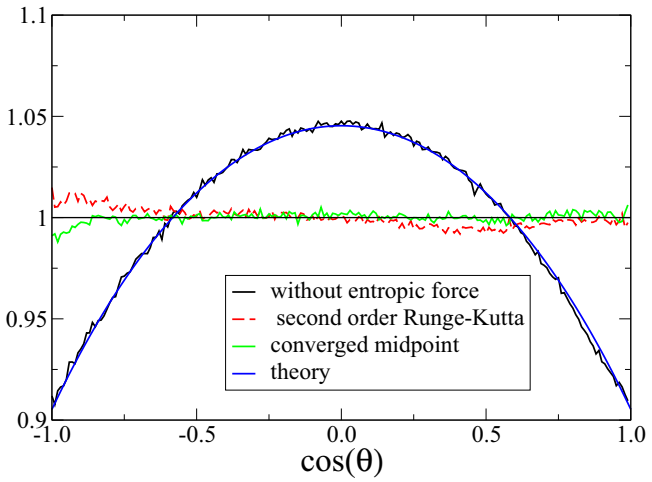

FIG. 2. Distribution of $\cos(\theta)$ for the trimer without the entropic force. Also drawn are the results from a midpoint resulting from a single iteration and of a converged iteration.

with one iteration computes the new point $\mathbf{u}'_a$ with the forward Euler scheme and takes for the midpoint

$$\mathbf{u}_a^{\text{mid}} = (\mathbf{u}'_a + \mathbf{u}_a)/2. \tag{48}$$

Then the forces and tensions are determined in this midpoint and with these values a new point $\mathbf{u}'_a$ is computed. This approximation is equivalent to a second order Runge-Kutta method and is recommended by Grassia *et al.* [9]. More iterations refine the determination of the midpoint.

The convergence of the iteration depends on the size of the time step: the larger the time step, the slower the convergence. We come back in detail to the optimum strategy in the next section. In Fig. 1 the distribution of the angle between the bonds of the trimer is plotted. The errors made in the forward Euler scheme, without iterations, are huge. The second curve is obtained with one iteration. We have used a time step $\Delta\tau = 0.001$; a smaller time step gives the same results.

On this scale one does not see differences between the single iteration and the flat (theoretical) distribution. In order to better appreciate the statistical errors we compare it in Fig. 2 with the distribution as it would follow by leaving out the entropic force, together with the theoretical prediction

$$P(\cos(\theta)) = \frac{3}{\pi + 1.5\sqrt{(3)}}\sqrt{4 - \cos^2(\theta)}. \tag{49}$$

On this scale, we see that second order Runge-Kutta method is not as perfect as it looked in the previous figure. For comparison we have also drawn the result of a converged iteration for the midpoint. The average number of iterations is 3, so the convergent midpoint routine needs a running time, which is twice that of second order Runge-Kutta method.

VI. LONGER CHAINS WITH RIGID BONDS

We now discuss polymer fragments which are semiflexible, i.e., of a length smaller than or of the order of the persistence length $l_p = J$. In order to keep the length of the fragment limited, we set $J = 120$, which is the persistence length of a dsDNA chain in units of the monomer distance. We consider chains of $N = 7$, 15, 31, 63, and 127 bonds, which gives a good impression of how the efficiency depends on the chain length. In this section we treat the chain with rigid bonds and study stiff bonds in the next section. The equations for a chain with rigid bonds are the same as those given in the previous section for the trimer, only the number of bonds is larger. That rules out an analytic solution for the tensions T_a, implying that tensions as well as the determinant of G and the entropic forces have to be calculated numerically. Since the matrix $G_{a,b}$ is tridiagonal all these calculations are of order N. As the update of the bonds is anyway an operation of order N, the calculation of the tensions and entropic forces slows down the calculation by a factor and not by a power of N.

The message is that it pays off to accurately determine a consistent midpoint. So for what follows we impose a criterion of convergence of the iteration and iterate as many times as is necessary to meet the criterion. With $\delta\mathbf{u}_n$ as the difference between two successive increments, we terminate the iteration if

$$\sum_n |\delta\mathbf{u}_n|^2 < 10^{-6}. \tag{50}$$

A sharper criterion 10^{-8} yields somewhat more precise midpoints, but has little influence on the values of the MSD of the middle monomer, which we take as main indicator of the quality of the simulation.

The average number of iterations gives a direct measure for the computer time needed for a time step. The time step divided by the average number of iterations yields the efficiency of the simulation. As the average number of iterations is already born out accurately by a short simulation, one can determine the most efficient time step before carrying out a long simulation. The ease with which the optimal time step can be determined is an indirect result of great convenience.

Typically we get the results as in Table I (for $N = 31$).

Here m is the number of iterations, $m_{\max}$ is the maximum number of iterations in the whole run and $\langle m\rangle$ is the average value. We have limited the number of iterations to 100, in order not to waste excessive time on midpoints which are difficult to determine accurately. This limit has been reached in the two largest time steps. The fourth column gives the efficiency of the time step. One observes that the average number of iterations to meet the criterion grows with the time-step size. The maximum efficiency is reached at the moment that the average number of iterations starts to grow faster than the time step. The physical quantities listed are the end-to-end distance

TABLE I. Maximal and average number of iterations of the iterative procedure ($m_{\max}$ and $\langle m\rangle$, respectively), as well as the resulting end-to-end distance L, the energy E and the MSD, for various time steps $\Delta\tau$.

$\Delta\tau$	$m_{\max}$	$\langle m\rangle$	$\Delta\tau/\langle m\rangle$	L	E	MSD
0.0001	7	3.76	0.0000266	29.743	29.7506	1.5325
0.0002	13	4.78	0.0000418	29.712	29.7501	1.5884
0.0003	44	6.05	0.0000495	29.723	29.7506	1.5699
0.0004	100	7.76	0.0000516	29.715	29.7506	1.5870
0.0005	100	10.38	0.0000481	29.705	29.7506	1.6045
Exact				29.717	29.7500	1.5820

L, the energy E, and the mean squared displacement (MSD) of the middle monomer.

Table I indicates that for $N = 31$ the optimum time step is near $\Delta\tau = 0.0004$, on the basis of the average number of needed iterations. While the average number of iterations is independent of the length of the simulation, the maximum occurring in a run, being determined by the most unfortunate configuration, increases with its length. So it is safer to simulate with a time step slightly below the optimum value, thereby avoiding a possible breakdown of the iteration procedure due to lack of convergence (or worse, divergence). For all chain lengths we can easily determine the optimal time step in a short simulation run. It turns out that it is fairly independent of the chain length and at the value $\Delta\tau = 0.0004$.

From the statistics of the jointed chain [13] we know the exact equilibrium values. As stated earlier, the most indicative value for the accuracy of the simulation is the MSD of the middle monomer. If that value comes out correctly, the energy and the end-to-end distance are more accurate by an order of magnitude. The table shows this clearly. In contrast to the statistics of the iterations one has to perform long simulations for good statistics on the MSD. We have verified that the midpoint routine gives accurate values of the MSD of the middle monomer, also for other values of N, if the optimal time step is used.

In order to get an idea how the midpoint algorithm compares with other solution schemes we have performed a number of simulations for the same values of N bonds, using the fourth order Runge-Kutta routine, which requires four evaluations of the forces in various points. The only way to find the efficiency of the Runge-Kutta method is to determine the largest allowable time step by inspecting the MSD of the middle monomer, being the most sensitive quantity for accuracy. For rigid bonds we know the equilibrium values. This works fine for short chains. E.g., for $N = 7$, we find that the MSD of the middle monomer deviates progressively from the exact value beyond the time step $\Delta\tau = 0.0004$, which is slightly smaller than the optimal time step $\Delta\tau = 0.0005$ for the midpoint routine. Since the midpoint routine takes on the average eight iterations for convergence, the Runge-Kutta integration is more efficient by a factor 2 for $N = 7$.

For longer chains the procedure to find the efficiency of the Runge-Kutta integration becomes cumbersome. The reason is the increasing relaxation times for longer chains. The longest relaxation time is due to the slowest transverse mode, which depends on N as [see Eq. (56)]

$$\tau_{\text{decay}} = (N+1)^4/(J\pi^4) \tag{51}$$

in units of our scaling. For $N = 31$ and $J = 120$ this is of the order 100. So with a time step 10^{-5} we need 10^7 times steps for one relaxation time. The MSD of the middle monomer strongly couples to the slowest mode. As the simulation itself increases with a factor N, the time needed for a relaxation time increases as N^5. In Fig. 3 we have plotted the ratio of the observed MSD of the middle monomer and the exact value, against N times the time step $\Delta\tau$. Since the point where the deviations from the exact value start to show up shifts to smaller time steps, one has to run the simulation longer by at least another factor N, bringing the power by which the simulation time increases

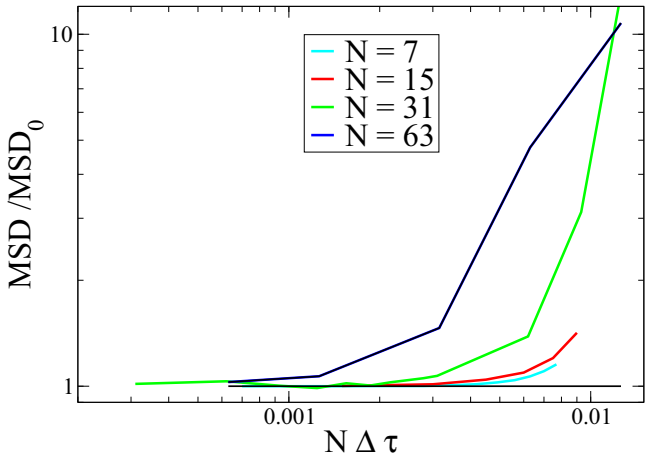

FIG. 3. Ratio of the observed MSD and the exact MSD_0 of the middle monomer for fourth order Runge-Kutta routine.

to faster than N^6. Therefore it was not possible to give reliable values for the Runge-Kutta method for $N = 127$.

The midpoint routine has the same problem with the intrinsic increase of N^5 for the simulation time, but the optimal time step remains fairly constant at the value $\Delta\tau = 0.0004$. Together with the average number of iterations (which is around 8) we can compare the efficiency of the midpoint routine with that of fourth order Runge-Kutta method. We find that below $N = 15$, Runge-Kutta method is more efficient and above $N = 15$ the midpoint routine wins. At $N = 63$ the ratio is about 20 in favor of the midpoint routine.

Checking detailed balance

The algorithm just discussed is equivalent with sampling the distribution (19) with the forces as derivatives of the free energy

$$\mathcal{F} = \mathcal{H} - \mathcal{S}. \tag{52}$$

As noted in Sec. III, one can change the algorithm such that it obeys detailed balance, without affecting the character of the dynamics. In the routine used above, one finds a new point by computing the force and tensions in the middle point. After this determination one can decide to accept or reject the new point on the basis of a second probability,

$$\mathcal{P}_2'(\Delta\mathbf{u}_n) \sim \exp\frac{1}{2}\left(\mathcal{F}(\mathbf{u}_n) + \sum_n \frac{\partial\mathcal{F}}{\partial\mathbf{u}_n}\cdot\Delta\mathbf{u}_n - \mathcal{F}(\mathbf{u}_n+\Delta\mathbf{u}_n)\right). \tag{53}$$

The value of $\mathcal{P}_2'$ has to be compared with a random number in the range R. Including this additional probability guarantees detailed balance and makes the simulation more accurate, but it does not lead to an increase in efficiency for the following reason. The exponent in Eq. (53) vanishes to quadratic order in the distance and for small time steps the exponent is close to zero, implying that the value of $\mathcal{P}_2$ is always close to 1. Then one can choose the range R of the random numbers close to 1, say $R = 1.1$. So about 10% of the moves will be rejected and the simulation is slowed down by the same percentage. If, however, for larger time steps, the range of values $\mathcal{P}_2'$ starts to

increase, one has to increase also the range R of the random numbers in order to exclude undersampling. Consequently the simulation gets slower by a factor $1/R$, without gain in efficiency.

We have checked the detailed balance condition in the above simulations. Near the optimum time step it is obeyed and beyond this value the range of $\mathcal{P}_2'$ starts to grow rapidly, which is another indication that one should not go beyond the optimum time step.

VII. CHAINS WITH STIFF BONDS

When a chain has stiff rather than rigid bonds it can be described by the Hamiltonian Eq. (11) with a finite value of Λ. In an earlier paper we have shown [15] that the integration becomes more efficient by using a mode representation, provided that the stretching stiffness is of the order of that of dsDNA. We discuss here this case, since it is an illustration of the option to separate off a (major) part of the Hamiltonian, which can be treated exactly. The mode representation that we use is in terms of Rouse modes R_p,

$$\mathbf{R}_p = \left(\frac{2}{N+1}\right)^{1/2} \sum_n \cos\left(\frac{(n+1/2)p\pi}{N+1}\right)\mathbf{r}_n, \tag{54}$$

and the zeroth order Hamiltonian reads

$$\mathcal{H}_0 = \frac{1}{2}\sum_p \zeta_p R_p^2. \tag{55}$$

We take as decay constants ζ_p

$$\zeta_p = 4J\left[1 - \cos\left(\frac{p\pi}{N+1}\right)\right]^2 \tag{56}$$

and show later on that it makes the remaining part $\Delta\mathcal{H}$ small. The Rouse modes are easier to handle than, e.g., the eigenfunctions of the biharmonic operator, which are the modes used by Munk *et al.* [20] for the IWLC [14].

The transformation (55) from positions to Rouse modes is an orthogonal transformation such that the Langevin equation in terms of Rouse modes reads

$$\frac{d\mathbf{R}_p}{d\tau} = -\zeta_p\mathbf{R}_p - \frac{\partial\Delta\mathcal{H}}{\partial\mathbf{R}_p} + \mathbf{G}_p. \tag{57}$$

Without $\Delta\mathcal{H}$ the equation describes an Ornstein-Uhlenbeck process, permitting an exact solution with the transition probability for the mode $\mathbf{R}_p'$ starting from a mode value $\mathbf{R}_p$ after a time interval $\Delta\tau$,

$$\mathcal{P}_1(\mathbf{R}_p') \sim \exp -\left(\frac{|\mathbf{R}_p' - \mathbf{R}_p\exp(-\zeta_p\Delta\tau)|^2}{2w_p^2}\right), \tag{58}$$

where the width w_p of the Gaussian is given by

$$w_p^2 = \frac{1-\exp(-2\zeta_p\Delta\tau)}{\zeta_p}. \tag{59}$$

In Appendix C we show that the transition probability Eq. (58) strictly obeys detailed balance.

A. The choice of ζ_p

$\Delta\mathcal{H}$ is implicitly defined as the difference of the Hamiltonian $\mathcal{H}$ given in Eq. (10) and the approximation $\mathcal{H}_0$ given in Eq. (55). Finding this difference requires a transformation from the modes back to the spatial representation in order to calculate $\mathcal{H}$. With this back transformation we find for $\mathcal{H}_0$ the expression

$$\mathcal{H}_0 = J\sum_n \left(u_n^2 - \mathbf{u}_n\cdot\mathbf{u}_{n+1}\right). \tag{60}$$

Comparing this with the definition Eq. (21) of the Hamiltonian yields $\Delta\mathcal{H}$ as

$$\Delta\mathcal{H} = \sum_n \left(\frac{\Lambda}{2}(u_n-1)^2 - Ju_n^2\right). \tag{61}$$

In order to make this expression more transparent we use the parameters

$$\nu = \frac{J}{\Lambda} \quad \text{and} \quad b = \frac{1}{1-2\nu}. \tag{62}$$

b is the length of the bond vector in the ground state. Then we get

$$\Delta\mathcal{H} = \frac{\Lambda}{2b}\sum_n (u_n - b)^2 + \frac{\Lambda N(1-b)}{2}. \tag{63}$$

This transparent form of the correction $\Delta\mathcal{H}$ is the justification of the choice Eq. (56) for ζ_p. $\Delta\mathcal{H}$ is a harmonic potential for the bond length around the ground-state bond length b. For large Λ one has $b \approx 1$, but b will differ substantially from 1 if Λ comes near $2J$ (the lower limit). The computation of $\Delta\mathcal{H}$ is easy in the spatial representation [and constant contributions drop out in Eq. (63)].

B. Refinement mode solution

In principle one could solve Eq. (57) by probing the additional probability

$$\mathcal{P}_2(\mathbf{R}_p') \sim \exp\tfrac{1}{2}[\Delta\mathcal{H}(\mathbf{R}_p) - \Delta\mathcal{H}(\mathbf{R}_p')]. \tag{64}$$

But this works only for very small times steps, since the energy difference tends to become large due to the steep Hamiltonian $\Delta\mathcal{H}$. Therefore we supplement the mode contribution to the step by the force $\mathbf{H}_p$ due to $\Delta\mathcal{H}$,

$$\mathbf{H}_p = -\frac{\partial\Delta\mathcal{H}}{\partial\mathbf{R}_p}. \tag{65}$$

Including the forces due to $\Delta\mathcal{H}$ in the probability $\mathcal{P}_1$ changes it into

$$\mathcal{P}_1'(\mathbf{R}_p') \sim \exp -\left(\frac{|\mathbf{R}_p' - \mathbf{R}_p e^{-\zeta_p\Delta\tau} - s_p\mathbf{H}_p|^2}{2w_p^2}\right). \tag{66}$$

In Appendix C it is shown that s_p has to have the form

$$s_p = \frac{1-e^{-\zeta_p\Delta\tau}}{\zeta_p}. \tag{67}$$

As earlier we want to compute the force $\mathbf{H}_p$ in the middle point, which is found through iteration. First we compute a new point $\mathbf{R}_p'$ using the force $\mathbf{H}_p$ of the old point $\mathbf{R}_p$. Then we determine in the newly found point the new force $\mathbf{H}_p'$ and

TABLE II. Optimum time step for a number of chain lengths N and values of the parameter ν.

ν	$N = 7$	$N = 15$	$N = 31$	$N = 63$
0.35	0.11	0.05	0.05	0.05
0.3	0.02	0.02	0.02	0.02
0.2	0.004	0.004	0.004	0.004
0.1	0.0004	0.0004	0.0004	0.0004

determine the middle value

$$\mathbf{H}_p^m = (\mathbf{H}_p + \mathbf{H}'_p)/2, \tag{68}$$

as a better approximation to the forces on the modes. Repeating this a number of times leads to a pair of consistent values of the midpoint force and computed new point. The iteration is considered convergent if the changes in midpoint obey the criterion (50). Then the probability (66) is symmetric in time and it obeys detailed balance within the approximation

$$\Delta\mathcal{H}' - \Delta\mathcal{H} + \sum_p \mathbf{H}_p \cdot (\mathbf{R}'_p - \mathbf{R}_p) \simeq 0. \tag{69}$$

In Table II below we give the optimal time steps for a number of chains and values of the parameter ν. The persistence length is kept on the value $J = 120$ (equaling the dsDNA value).

Two aspects are noteworthy: the optimal time step is hardly dependent on the length of the chain and decreases very rapidly with ν. The latter feature is clear, since a smaller ν means a stiffer bond (as J is fixed) and so smaller time steps are needed in the steep potential. The independence of the chain length is comparable to the midpoint algorithm for rigid bonds.

For values below $\nu = 0.1$ the use of modes becomes less efficient compared to the rigid bond algorithm, which uses only the spatial representation.

In an earlier paper [15] we have estimated the maximum allowable time step for dsDNA on the basis of the behavior of the time-dependent correlation functions. When properly translated we find comparable values with the first row in the table. The earlier found values are the results of many long simulations since we had only the independence of the quantities on the time step available. Presently the values in the table follow from short runs as the average number of iterations is already born out by a short run.

C. Checking detailed balance

Complete detailed balance can be enforced by sampling as a second probability, instead of Eq. (66),

$$\mathcal{P}'_2(\mathbf{R}'_p) \sim \exp -\frac{1}{2}\left(\Delta\mathcal{H}' - \Delta\mathcal{H} + \sum_p \mathbf{H}_p \cdot (\mathbf{R}'_p - \mathbf{R}_p)\right). \tag{70}$$

One would hope that the exponent of $\mathcal{P}'_2$ is close to zero, such that the probabilities stay close to 1. Then a range in random numbers slightly larger than 1 would suffice. For short chains and larger ν this is indeed the case, but for longer chains the range of values increases and one loses more time with the implementation of $\mathcal{P}'_2$ by an accept-or-reject routine than one gains in accuracy.

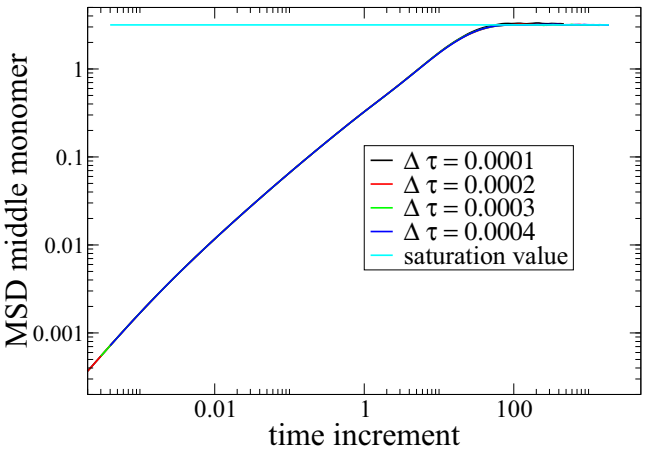

FIG. 4. The MSD correlation function of the middle monomer for a series of time steps for a chain with $N = 31$ bonds.

VIII. TIME DEPENDENT QUANTITIES

As stated before the time correlation functions are usually correct if they evolve to their correct saturation values. We inspect this explicitly for the time correlation of the MSD of the middle monomer, which is given in Eq. (15). It starts as a diffusion process in a linear way and saturates to twice the equilibrium value, since for large time differences the correlation with the initial value vanishes.

In Fig. 4 we have drawn the values of $C(\tau)$ for the same series of time steps given in Sec. VI for the chain with $N = 31$ bonds. We see that the curves for the various time steps are on top of each other and approach the saturation value as given by the theory for rigid bonds. This shows that indeed the temporal behavior of the correlation functions is independent of the size of the time steps as long as the time steps are below the optimum value.

Note that the decay time $\tau_{\text{decay}} \simeq 100$, as following from Fig. 4, corresponds to the estimate Eq. (51) for $N = 31$.

IX. DISCUSSION

We have been studying the equations of motion for polymers with extensible (stiff) and inextensible (rigid) monomer distance. By reconsidering the derivations we conclude that the proper limit from stiff to rigid bonds requires only an extra entropic force (pseudopotential) and not a correction due to a varying diffusivity. The origin of the extra force is not a higher order effect of random forces acting in a constrained subspace, as suggested in [7,9], but a residual phase space effect (entropy) of the limit of stiff to rigid bonds.

We have investigated bead-spring chains with bonds varying from stiff to rigid, with two questions in mind: (i) Is the algorithm for solution of the stochastic (Langevin) equation "in principle exact"? (ii) How efficient is the algorithm?

The answer to the first question is demonstrated at the hand of the trimer. Straightforward Euler integration gives wrong results no matter how small one takes the time step. Higher order schemes, such as, e.g., the second order Runge-Kutta method, are in principle exact, but require small time steps and correspondingly long simulations for an accurate simulation.

We have worked out a solution of the Langevin equation based on a consistent pair of a final configuration and the midpoint between initial and final configuration, such that the forces in the midpoint yield the final configuration by linear interpolation. Such a pair is obtained through iteration. The number of iterations needed for convergence grows with the size of the time step and slows down the simulation accordingly, since the determination of the forces is a major part of the implementation of the time step. The efficiency of the simulation is the ratio of the time step and the average number of iterations. The optimal time step occurs at the maximum of the efficiency. The optimum is already born out by a short simulation, enabling us to choose the optimum time step before long simulations are carried out. The necessity of long simulations is intrinsically due to the long relaxation times of the chain, that grow as N^4 with the chain length N. We have shown that a sufficiently long simulation at the optimal time step gives the correct value for the MSD of the middle monomer and *a fortiori* those of other properties like the end-to-end distance and the energy.

We have determined the optimum efficiency for rigid and stiff bonds. For rigid bonds we have compared the midpoint routine with the standard fourth order Runge-Kutta method, which consumes as much time per step as four iterations. The maximum allowable time step in the Runge-Kutta solutions has been determined by locating the point where the MSD of the middle monomer deviates substantially from the exact known value. Thus we find that the Runge-Kutta method is only efficient for short chains (less than $N = 15$), since the maximum allowable time step goes rapidly down with the length of the chain, while for the midpoint routine the optimum time step is hardly dependent on the length of the chain. The ratio of the efficiency of the midpoint routine and fourth order Runge-Kutta method has been estimated, but this estimate is hampered by the fact that accurate simulations using the Runge-Kutta method require excessively long simulations for longer chains, since the allowable time step becomes too small for longer chains, while the relevant relaxation times rapidly grow with the length of the chain.

For rigid bonds the midpoint routine obeys detailed balance for time steps below the optimum. For stiff bonds one could enforce detailed balance, but that would make the routine inefficient, as most of the moves would have to be rejected. In spite of this shortcoming, the midpoint routine gives very good averages for the optimum time step, for a stiffness of the order of that of dsDNA. The optimal time step is more than two orders larger than that of rigid bonds.

The optimal time step goes rapidly down for increasing stiffness. The point has been determined where the optimum time step becomes of the order of that for rigid bonds.

We have verified the claim that correct simulation of the equilibrium properties (saturation values), in a long run, is a guarantee that the time dependent correlation functions also come out right for the whole evolution.

ACKNOWLEDGMENTS

We gratefully acknowledge D. Panja for useful discussions, in particular for pointing us to some of the references in the mathematical literature, and M. Pasquali and D. Morse for illuminating discussions on their work.

APPENDIX A: THE ENTROPIC FORCE

For the derivation of the entropic force we return to the Hamiltonian (1) for very large λ and study the motion of the fluctuations

$$\delta u_a = u_a - 1. \tag{A1}$$

The time derivative of the fluctuation follows as

$$\delta\dot{u}_a = \frac{du_a}{d\mathbf{u}_a}\cdot\dot{\mathbf{u}}_a. \tag{A2}$$

Using the equations for the time derivative $\dot{\mathbf{u}}_a$ we find

$$\delta\dot{u}_a = -\lambda\sum_b G_{a,b}\delta u_b + \cdots, \tag{A3}$$

where the dots stand for terms remaining finite as $\lambda\to\infty$. They may be ignored in view of the size of the given terms. $G_{a,b}$ is the same matrix as introduced in Eq. (34). The fluctuations carry out excursions around the subspace. In order to find the width of these excursions we must diagonalize the matrix G by the orthogonal transformation

$$\delta w_a = \sum_b \phi_{a,b}\delta u_b \tag{A4}$$

leading to the eigenvalue equations

$$\sum_b G_{a,b}\phi_{b,c} = \gamma_c\phi_{a,c}. \tag{A5}$$

Then the equations of motion for the δw_a read

$$\delta\dot{w}_a = -\lambda\gamma_a\delta w_a + \cdots. \tag{A6}$$

We thus find for the average squared fluctuations ($k_BT = 1$ in our units)

$$\langle(\delta w_a)^2\rangle = \frac{1}{\lambda\,\gamma_a}. \tag{A7}$$

The factor λ makes the magnitude of the fluctuations very small. The eigenvalues tell how much this width depends on the configuration. Therefore the overall width of the zone of fluctuations around the constrained value becomes

$$\text{width} = \prod_a \langle(\delta w_a)^2\rangle^{1/2} \sim \prod_a (\gamma_a)^{-1/2} = 1/\sqrt{\det G}. \tag{A8}$$

Identifying the width with the density of states, we obtain the same expression for the entropic force as given in Eq. (37). Inspecting the line of arguments one sees that the correlation between the fluctuations of the bond lengths is the origin of the entropic force. Without the off-diagonal elements in the matrix G all bonds would fluctuate with the same frequency and the entropic force would vanish.

APPENDIX B: HIGHER ORDER STRESSES

In this appendix we show how to derive the expression Eq. (35) of Hinch in the simple case of equal masses and friction coefficients. It follows from higher orders in the stresses due to the curvature of the constraints, leading to an

additional drift in the motion. In order to shorten the notation we introduce the abbreviations

$$\mathbf{u}_n^a = \frac{\partial u_a}{\partial \mathbf{r}_n}, \quad \mathbf{u}_{nm}^a = \frac{\partial^2 u_a}{\partial \mathbf{r}_n \partial \mathbf{r}_m} \tag{B1}$$

and employ the summation convention for repeated indices. As the correlation only matters for the random forces, we leave out here the systematic forces, such that the equation of motion becomes

$$\dot{\mathbf{r}}_n = \mathbf{g}_n + T_a \mathbf{u}_n^a. \tag{B2}$$

The velocity constraint then reads

$$0 = \mathbf{u}_n^a \cdot \dot{\mathbf{r}}_n = \mathbf{u}_n^a \cdot \mathbf{g}_n + G_{ab} T_b, \tag{B3}$$

with

$$G_{ab} = \mathbf{u}_n^a \cdot \mathbf{u}_n^b. \tag{B4}$$

Solving the tension from Eq. (B3) we get the equation of motion

$$\dot{\mathbf{r}}_n = \mathbf{g}_n - \mathbf{u}_n^a \, G_{a,b}^{-1} \, \mathbf{u}_m^b \cdot \mathbf{g}_m \equiv \mathbf{P}_{n,m} \cdot \mathbf{g}_m, \tag{B5}$$

where $\mathbf{P}_{n,m}$ is the geometric projection operator.

The idea is that there is a first order tension $T_a^{(1)}$ associated with a first order displacement $\mathbf{r}_n^{(1)}$, following from the first order equation (B5) and in addition a second order tension $T_a^{(2)}$, which modifies the velocity,

$$\begin{aligned} \dot{\mathbf{r}}_n^{(1)} &= \mathbf{g}_n + T_a^{(1)} \, \mathbf{u}_n^a, \\ \dot{\mathbf{r}}_n^{(2)} &= T_a^{(2)} \, \mathbf{u}_n^a + T_a^{(1)} \, \mathbf{r}_m^{(1)} \cdot \mathbf{u}_{nm}^a. \end{aligned} \tag{B6}$$

The first equation vanishes upon averaging and in order to work out the average of the second equation, we start with a computation of the diffusivity $\mathbf{D}_{mn}$

$$\mathbf{D}_{mn} = \left\langle \dot{\mathbf{r}}_n^{(1)} \mathbf{r}_m^{(1)} \right\rangle, \tag{B7}$$

which is obtained by multiplying the first Eq. (B6) with $\mathbf{r}_m^{(1)}$ and taking the average

$$\mathbf{D}_{mn} = \left\langle \mathbf{g}_n \mathbf{r}_m^{(1)} \right\rangle + \left\langle T_a^{(1)} \mathbf{r}_m^{(1)} \right\rangle \mathbf{u}_n^a. \tag{B8}$$

The first term on the right-hand side is obtained by using the equation of motion Eq. (B5) for a short time interval τ

$$\begin{aligned} &\left\langle \mathbf{g}_n(\tau) \mathbf{r}_m^{(1)}(\tau) \right\rangle \\ &= \left\langle \mathbf{g}_n(\tau) \int_0^\tau d\tau' \left[\mathbf{g}_m(\tau') - G_{a,b}^{-1} \, \mathbf{u}_m^a \, \mathbf{u}_k^b \cdot \mathbf{g}_k(\tau') \right] \right\rangle. \end{aligned} \tag{B9}$$

The random forces at different times are δ correlated as

$$\langle \mathbf{g}_n(\tau) \mathbf{g}_m(\tau') \rangle = 2\delta(\tau - \tau') \delta_{n,m} \tag{B10}$$

and therefore we obtain

$$\left\langle \mathbf{g}_n(\tau) \mathbf{r}_m^{(1)}(\tau) \right\rangle = \mathbf{I} \, \delta_{n,m} - \mathbf{u}_n^a \, G_{a,b}^{-1} \, \mathbf{u}_m^b = \mathbf{P}_{n,m}. \tag{B11}$$

So we get for the diffusivity the expression

$$\mathbf{D}_{mn} = \mathbf{P}_{n,m} + \left\langle T_a^{(1)} \mathbf{r}_m^{(1)} \right\rangle \mathbf{u}_n^a. \tag{B12}$$

The projection operator has the property that it is perpendicular to the gradient of the constraint

$$\mathbf{u}_n^c \cdot \mathbf{P}_{n,m} = \mathbf{u}_m^c - G_{c,a} G_{a,b}^{-1} \, \mathbf{u}_m^b = \mathbf{u}_m^c - \mathbf{u}_m^c = 0. \tag{B13}$$

Similarly the velocity constraint implies that

$$\mathbf{u}_n^c \cdot \mathbf{D}_{n,m} = 0. \tag{B14}$$

So, if we apply the same operation to Eq. (B12), we find that the cross correlations between the first order tensions and displacements vanish,

$$0 = G_{b,a} \left\langle T_a^{(1)} \mathbf{r}_m^{(1)} \right\rangle, \tag{B15}$$

since these homogeneous linear equations have only the zero solution. This has two consequences: the diffusivity equals the projection operator

$$\mathbf{D}_{n,m} = \mathbf{P}_{n,m} = \mathbf{I} \, \delta_{n,m} - \mathbf{u}_n^a \, G_{a,b}^{-1} \, \mathbf{u}_m^b \tag{B16}$$

and the extra average drift reduces to

$$\left\langle \dot{\mathbf{r}}_n^{(2)} \right\rangle = \left\langle T_a^{(2)} \right\rangle \mathbf{u}_n^a. \tag{B17}$$

The equation for $\langle T_a^{(2)} \rangle$ is obtained by considering the second order terms of the velocity constraint (B4),

$$0 = \mathbf{u}_n^a \left\langle \dot{\mathbf{r}}_n^{(2)} \right\rangle + \mathbf{u}_{nm}^a \left\langle \mathbf{r}_m^{(1)} \dot{\mathbf{r}}_n^{(1)} \right\rangle, \tag{B18}$$

which yields Eqs. (B7) and (B17),

$$0 = G_{a,b} \left\langle T_b^{(2)} \right\rangle + \mathbf{u}_{nm}^a \, \mathbf{D}_{n,m}. \tag{B19}$$

So we may solve for $\langle T_a^{(2)} \rangle$ and insert it into Eq. (B17) with the "final" result

$$\left\langle \dot{\mathbf{r}}_n^{(2)} \right\rangle = -\mathbf{u}_n^a \, G_{a,b}^{-1} \mathbf{u}_{km}^b \, \mathbf{D}_{k,m}, \tag{B20}$$

which works out to the expression

$$\left\langle \dot{\mathbf{r}}_n^{(2)} \right\rangle = -\mathbf{u}_n^a \cdot G_{a,b}^{-1} \, \mathbf{u}_{mm}^b + \mathbf{u}_n^a \, G_{a,b}^{-1} \mathbf{u}_{km}^b \, \mathbf{u}_k^c \, G_{c,d}^{-1} \mathbf{u}_m^d. \tag{B21}$$

This is an explicit equation for the second order drift, but it comes in a form which is not very transparent.

In order to see that this expression is the same as that of Hinch we first remark that the determinant of G is the product of the eigenvalues γ_α,

$$\det G = \prod_\alpha \gamma_a. \tag{B22}$$

Differentiating the product with respect to a coordinate $\mathbf{r}_n$ leads to

$$\frac{\partial \det G}{\partial \mathbf{r}_n} = \det G \sum_\alpha \frac{1}{\gamma_\alpha} \frac{\partial \gamma_\alpha}{\partial \mathbf{r}_n}. \tag{B23}$$

Using the representation of the matrix $G_{a,b}$ as

$$G_{a,b} = \sum_\alpha \phi_a^\alpha \, \gamma_\alpha \, \phi_b^\alpha \tag{B24}$$

and using the orthonormality of the eigenfunctions ϕ_a^α one finds

$$\frac{\partial \ln(\det G)}{\partial \mathbf{r}_n} = G_{a,b}^{-1} \frac{\partial}{\partial \mathbf{r}_n} G_{b,a}. \tag{B25}$$

On the other hand a direct differentiation of the matrix elements yields

$$\frac{\partial}{\partial \mathbf{r}_n} G_{b,a} = \left(\mathbf{u}_{nm}^b \cdot \mathbf{u}_m^a + \mathbf{u}_m^b \cdot \mathbf{u}_{nm}^a \right). \tag{B26}$$

Inserting this into Eq. (B25) gives

$$\frac{\partial \ln(\det G)}{\partial \mathbf{r}_n} = \mathbf{u}_m^a \cdot G_{a,b}^{-1} \, \mathbf{u}_{nm}^b + \mathbf{u}_{nm}^a \cdot G_{a,b}^{-1} \, \mathbf{u}_m^b. \tag{B27}$$

Note that the two terms are equal since G is symmetric and thus we may interchange the summations over a and b. We keep the last and project on to the subspace yielding

$$\frac{1}{2}\mathbf{P}_{nm}\frac{\partial \ln(\det G)}{\partial \mathbf{r}_m} = \mathbf{u}^a_m \cdot G^{-1}_{a,b}\,\mathbf{u}^b_{nm} - \mathbf{u}^a_n G^{-1}_{a,b}\,\mathbf{u}^b_m\,\mathbf{u}^c_{mk} G^{-1}_{cd}\,\mathbf{u}^d_k. \tag{B28}$$

This brings the entropic term of the expression Eq. (35) in a convenient form.

The diffusivity term in Eq. (35) leads to three contributions

$$-\frac{\partial}{\partial \mathbf{r}_m}\cdot \mathbf{D}_{n,m} = \mathbf{u}^a_{nm}\cdot G^{-1}_{a,b}\,\mathbf{u}^b_m + \mathbf{u}^a_n \cdot \frac{\partial G^{-1}_{a,b}}{\partial \mathbf{r}_m}\,\mathbf{u}^b_m + \mathbf{u}^a_n \cdot G^{-1}_{a,b}\,\mathbf{u}^b_{mm}. \tag{B29}$$

We already see terms that correspond and that cancel. The first term of Eq. (B28) cancels the first term in Eq. (B29) and the last term in Eq. (B29) equals the first term in our result Eq. (B21). So we continue with the middle term of Eq. (B29) and use

$$\frac{\partial G^{-1}_{a,b}}{\partial \mathbf{r}_m} = -G^{-1}_{a,c}\frac{\partial G_{c,d}}{\partial \mathbf{r}_m}G^{-1}_{d,b}. \tag{B30}$$

This term is worked out with Eq. (B26) to

$$-\mathbf{u}^a_n \cdot \frac{\partial G^{-1}_{a,b}}{\partial \mathbf{r}_m}\,\mathbf{u}^b_m = \mathbf{u}^a_n\, G^{-1}_{a,b}\left(\mathbf{u}^b_{km}\,\mathbf{u}^c_m + \mathbf{u}^b_m\,\mathbf{u}^c_{mk}\right) G^{-1}_{c,d}\,\mathbf{u}^d_k. \tag{B31}$$

Now one sees that the first term on the right-hand side equals the second term in our expression Eq. (B21) and that the second term cancels the second term in Eq. (B28). This completes the proof that our result Eq. (B21) is equivalent with the expression Eq. (35) of Hinch.

APPENDIX C: DETAILED BALANCE FOR INDEPENDENT MODES

We show in this section that the transition probability Eq. (33) rigorously obeys detailed balance. For clarity we write it as

$$\mathcal{P}(\mathbf{R}_p \to \mathbf{R}'_p) \sim \exp -\left(\frac{|\mathbf{R}'_p - \mathbf{R}_p e^{-\zeta_p \Delta\tau}|^2}{2w_p^2}\right). \tag{C1}$$

So the reverse transition has the probability

$$\mathcal{P}(\mathbf{R}'_p \to \mathbf{R}_p) \sim \exp -\left(\frac{|\mathbf{R}_p - \mathbf{R}'_p e^{-\zeta_p \Delta\tau}|^2}{2w_p^2}\right). \tag{C2}$$

The ratio equals

$$\frac{\mathcal{P}(\mathbf{R}_p \to \mathbf{R}'_p)}{\mathcal{P}(\mathbf{R}'_p \to \mathbf{R}_p)} = \exp -\left(\frac{\left(R'^2_p - R^2_p\right)(1 - e^{-2\zeta_p\Delta\tau})}{2w_p^2}\right). \tag{C3}$$

Using the definition Eq. (34) of the width we get

$$\frac{\mathcal{P}(\mathbf{R}_p \to \mathbf{R}'_p)}{\mathcal{P}(\mathbf{R}'_p \to \mathbf{R}_p)} = \exp -\left(\frac{1}{2}\zeta_p\left(R'^2_p - R^2_p\right)\right). \tag{C4}$$

The argument of the exponential on the right-hand side is indeed the difference of the energies of the modes, showing that detailed balance is strictly obeyed.

One wonders what the inclusion of the coupling force in the probability Eq. (C1) does to detailed balance. So consider instead of Eq. (C1) the probability

$$\mathcal{P}(\mathbf{R}_p \to \mathbf{R}'_p) \sim \exp -\left(\frac{|\mathbf{R}'_p - \mathbf{R}_p e^{-\zeta_p \Delta\tau} - s_p \mathbf{H}_p|^2}{2w_p^2}\right), \tag{C5}$$

where $\mathbf{H}_p$ is the force due to $\Delta\mathcal{H}$,

$$\mathbf{H}_p = -\frac{\partial \Delta\mathcal{H}}{\partial \mathbf{R}_p}, \tag{C6}$$

and s_p is the factor

$$s_p = \frac{1 - e^{-\zeta_p\Delta\tau}}{\zeta_p}. \tag{C7}$$

s_p follows from the time-integrated Langevin equation. Now suppose that we have managed to construct this force at a middle point such that we may use the same $\mathbf{H}_p$ for the backward motion with the probability

$$\mathcal{P}(\mathbf{R}'_p \to \mathbf{R}_p) \sim \exp -\left(\frac{|\mathbf{R}_p - \mathbf{R}'_p \mathrm{e}^{-\zeta_p \Delta\tau} - s_p \mathbf{H}_p|^2}{2w_p^2}\right). \tag{C8}$$

Then it is an algebraic exercise to show that again detailed balance is fulfilled using the approximation

$$\Delta\mathcal{H}' - \Delta\mathcal{H} + \sum_p \mathbf{H}_p \cdot (\mathbf{R}'_p - \mathbf{R}_p) \simeq 0. \tag{C9}$$

One can use the left-hand side of this equation as the criterion for the acceptance probability in order to reinstall full detailed balance.

[1] H. A. Kramers, J. Chem. Phys. **14**, 415 (1947).
[2] E. Helfand, J. Chem. Phys. **71**, 5000 (1979).
[3] N. G. van Kampen and J. J. Lodder, Am. J. Phys. **52**, 419 (1984).
[4] M. Fixman, J. Chem. Phys. **69**, 1527 (1974).
[5] D. C. Morse, Theory of constrained Brownian motion, in *Advances in Chemical Physics*, Vol. 128, edited by S. A. Rice (John Wiley & Sons, Inc., Hoboken, NJ, USA, 2003).
[6] A. Montesi, D. C. Morse, and M. Pasquali, J. Chem. Phys. **122**, 084903 (2005).
[7] J. E. Hinch, J. Fluid Mech. **271**, 219 (1994).
[8] D. L. Ermak and J. A. McCammon, J. Chem. Phys. **69**, 1352 (1979).
[9] P. S. Grassia, E. J. Hinch, and L. C. Nitsche, J. Fluid Mech. **282**, 373 (1995).
[10] Chien-Cheng Chang, Math. Comput. **49**, 523 (1987).
[11] J. Wilkie, Phys. Rev. E **70**, 017701 (2004).
[12] M. D. Wang, H. Yin, R. Landick, J. Gelles, and S. M. Block, Biophys. J. **72**, 1335 (1997).
[13] G. T. Barkema and J. M. J. van Leeuwen, J. Stat. Mech. (2012) P12019.

[14] G. T. Barkema, D. Panja and J. M. J. van Leeuwen, J. Stat. Mech. (2014) P11008.
[15] D. Panja, G. T. Barkema, and J. M. J. van Leeuwen, Phys. Rev. E **92**, 032603 (2015).
[16] P. S. Lang, B. Obermayer, and E. Frey, Phys. Rev. E **89**, 022606 (2014).
[17] B. Obermayer and E. Frey, Phys. Rev. E **80**, 040801 (2009).
[18] M. E. J. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics* (Oxford University Press, Oxford, 1999), p. 406.
[19] We use the definition of Morse *et al.* rather than that of Hinch who defines G^{-1} by Eq. (32).
[20] T. Munk, O. Hallatschek, C. H. Wiggins, and E. Frey, Phys. Rev. E **74**, 041911 (2006).