Calculating Hydrodynamic Properties of DNA through a Second-Order Brownian Dynamics Algorithm

Giuseppe Chirico† and Jörg Langowski*

EMBL, Grenoble Outstation, 156X, F-38042 Grenoble Cedex, France

Received June 24, 1991; Revised Manuscript Received October 7, 1991

ABSTRACT: We describe the application of a second-order Brownian dynamics algorithm (Iniesta, A.; Garcia de la Torre, J. *J. Chem. Phys.* **1990**, *92* (3), 2015) to the calculation of hydrodynamic properties of simple bead–chain models of linear DNA, consisting of 2–17 circular beads of a diameter of 3.184 nm. Hydrodynamic interactions were incorporated using the Rotne–Prager tensor (Rotne, J.; Prager, S. *J. Chem. Phys.* **1969**, *50* (11), 4831). The results of the computations were tested against properties of the model chains that could be calculated from other means, such as end-to-end distance, diffusion coefficient, or rotational relaxation time. Typical trajectory lengths were up to 11.4 μs for a 10 subunit (93 base pair) chain, which allowed us to predict the DLS structure factor to a maximum time of 1–2 μs. Compared with a first-order algorithm (Ermak, D. L.; McCammon, J. A. *J. Chem. Phys.* **1978**, *69* (4), 1352), the second-order calculation achieves the same accuracy in 4–5 times less CPU time.

I. Introduction

The conformation and dynamics of the DNA chain has become a subject of major interest in biophysics in recent years. In addition, DNA constitutes an ideal model for polymer physics to study the properties of wormlike chains, since monodisperse preparations of milligram amounts of linear or circular DNAs can easily be obtained through biochemical and molecular biology techniques.[1]

A circular DNA chain can be torsionally stressed, since the DNA double helix cannot rotate freely around its axis. Such torsional stress causes the DNA to assume a superhelical conformation, in which two DNA double helices are wound around each other.[2,3] Naturally occurring DNA is often subject to torsional stress and is therefore superhelical. It has been shown in several instances that DNA superhelicity influences DNA function in various ways: by changing local conformation through the internal torsional stress, or by changing the mutual arrangement of interacting DNA sequences. For understanding processes like intracellular DNA rearrangements or protein binding to specific DNA sequences, it is important to understand the structure *and* internal motions of linear and superhelical DNAs.[2]

There exists a wealth of data on DNA internal motions, measured with methods such as dynamic light scattering (DLS), fluorescence polarization anisotropy (FPA), NMR, electric birefringence, or electric dichroism, to name only a few. Our own work has mainly dealt with studies of DNA internal motions by DLS.[4,5] The advantage of DLS in studying a large flexible macromolecule like DNA is that the method is noninvasive and does not perturb the equilibrium conformation of the molecule. The drawback is that the information gained from DLS cannot be easily connected to basic physical properties of the DNA chain. The only parameters directly accessible are the translational (D_t) and rotational (Θ) diffusion coefficients of a semiflexible rodlike molecule, from DLS measurements at small scattering vectors $\mathbf{K}$ with $|\mathbf{K}|R_g < 1$, R_g being the radius of gyration of the molecule.[6,7] For measurements at larger $\mathbf{K}$ no complete theory exists that would predict the dynamic structure factor $S(\mathbf{K},t)$. Several analytical approximations have been proposed,[8-10] but none of these bear any prospect to include "real-world" parameters of DNA like sequence-dependent torsional and bending elasticity, bending anisotropy, permanent curvature, or the torsional constraints that determine the structure of superhelical DNA.

Presently, the only way to connect DLS measurements on DNA to fundamental physical parameters of the molecule is through a numerical simulation of its conformation and dynamics. Some applications of Brownian dynamics (BD) to studies of DNA internal motions have already been described, notably through the work of Allison and co-workers.[11-14,26] Those simulations have been confined to linear DNA chains. Inclusion of the torsional constraints is possible with the algorithm described there; an example has been given in the recent calculation of the transient photodichroism of a 209 base pair DNA fragment.[26] However, with increasing number of chain subunits, and especially for chain lengths of several thousand base pairs as in the case of superhelical DNA, computation time rises prohibitively.

We describe in the following a Brownian dynamics model based on a second-order algorithm described by Iniesta and Garcia de la Torre.[15] This method speeds up the computation sufficiently that simulations on torsionally constrained circular DNA chains come within reach. In this paper, we describe the implementation of the algorithm, and comparative tests on various models with the earlier first-order procedure as used by Allison.[11-14] All calculations reported here still use a bending–stretching potential; the torsional potential has been implemented and will be the subject of a follow-up publication.

II. Theory

a. General. The Brownian dynamics (BD) procedure that we are going to develop in the following is aimed at modeling the dynamical properties of medium-size DNA fragments (several 100 base pairs) that can be measured by methods such as dynamic light scattering (DLS) or fluorescence polarization anisotropy decay (FPA). The time scale operated on by these methods is in the nanosecond (FPA) to micro-to millisecond (DLS) range, so that atomic scale dynamic simulations would not be tractable on today's computing equipment.

Since the properties we are interested in are not a priori affected by local conformation, we model the DNA molecule as a chain of beads; in the simulations presented

* To whom reprint requests should be sent.
† Present address: Sezione di stati aggregati, Dipartimento di Fisica, Università di Milano, Via Celoria 16, I-20133 Milano, Italy.

0024-9297/92/2225-0769$03.00/0

in this paper we choose the bead size and interbead distance such that the chain is contiguous and its hydrodynamic diameter has the known value of 1.25 nm. It has been shown[16] that correct hydrodynamic parameters for DNA fragments are reproduced when the bead diameter d is 2σ = 3.184 nm, the total chain volume being equal to that of a DNA of equal length. Thus, one bead equals 9.3 base pairs of DNA. The number N of beads is given by the known length of the DNA fragment and the bead diameter.

The coordinates of bead j are $\mathbf{r}_j$ in the laboratory frame of reference. These beads are connected by elastic joints with a given bending rigidity. Since DNA is very stiff toward longitudinal deformations, the distance between beads should be kept constant, or close to an average constant value. This can be achieved either by imposing a strong stretching potential, or as in ref 11 through the SHAKE algorithm. For reasons of computational convenience, we chose the route of the stretching potential; in the course of the paper it is shown that this gives very satisfactory results even for stiff arrangements of beads.

The stretching potential U_s is defined as

$$U_s = \frac{k_B T}{2\delta^2} \sum_{j=1}^{N-1} (|\mathbf{b}_j| - b_0)^2 \tag{1}$$

where $k_B T/2\delta^2$ is the force constant of the stretching potential, b_j is equal to $|\mathbf{b}_j| = |\mathbf{r}_{j+1} - \mathbf{r}_j|$, i.e. to the jth bond length, and b_0 is the equilibrium interbead distance 2σ = 3.184 nm.

The bending potential U_b has the form

$$U_b = \frac{k_B T}{2\gamma^2} \sum_{j=1}^{N-2} \alpha_j^2 \tag{2}$$

where $k_B T/2\gamma^2$ is the force constant of the bending potential. The $\{\alpha_j\}_{j=1}^{N-2}$ are the $N-2$ bending angles defined by the relation

$$\cos(\alpha_j) = \frac{\mathbf{r}_{j+1,j+2} \cdot \mathbf{r}_{j,j+1}}{b_{j+1} b_j} \tag{3}$$

where $\mathbf{r}_{j,j+1}$ is the bond vector from bead j to $j+1$, and b_j its length.

In this paper, we only deal with linear DNA fragments without permanent bends. In that case, simulating the DLS data only requires the use of a bending potential, since the torsional motions have decayed on the DLS time scale, and bending and torsional fluctuations are statistically independent, as has been shown in ref 17. The influence of a torsion potential between beads will be studied in a later paper, particularly in the view of simulating the dynamics of torsionally stressed circular (superhelical) DNA.

The simulation of the DLS decay of DNA fragments of several hundred base pairs, even with a bead–chain model, is a very computation-intensive task. In order to achieve longer simulation times, for the same CPU time and precision, than the Ermak–McCammon algorithm used in previous work of this kind,[18] a BD algorithm of higher accuracy has been proposed recently.[15] We shall briefly review this algorithm here.

In the Langevin equation for the translational motion of a polymer consisting of N beads of equal mass m in an incompressible continuum solvent:

$$m\frac{d\mathbf{v}_i}{dt} = -\sum_{j=1}^{N} \zeta_{ij} \cdot \mathbf{v}_j + \mathbf{F}_i + \sum_{j=1}^{N} \alpha_{ij} \cdot \mathbf{f}_j \tag{4}$$

$\mathbf{v}_i$ is the velocity of the ith bead, and $\mathbf{F}_i$ is the sum of the external and interparticle forces acting on it at time t. The ζ_{ij} are the translational friction tensors which are related to the translational diffusion tensors $\mathbf{D}_{ij} = k_B T (\zeta_{ij})^{-1}$. α_{ij} indicates the stochastic term. $\mathbf{D}_{ij}$ is the Rotne–Prager tensor:

$$\mathbf{D}_{ij} = \frac{k_b T}{6\pi\eta\sigma} \mathbf{I} \qquad i = j$$

$$\mathbf{D}_{ij} = \frac{k_b T}{8\pi\eta b_{ij}} \left\{ \mathbf{I} + \left[\frac{\mathbf{r}_{ij} : \mathbf{r}_{ij}}{b_{ij}^2} \right] + \left(\frac{2\sigma^2}{3b_{ij}^2} \right) \left[\mathbf{I} - 3\left(\frac{\mathbf{r}_{ij} : \mathbf{r}_{ij}}{b_{ij}^2} \right) \right] \right\} \qquad i \neq j \tag{5}$$

for different but nonoverlapping beads, and

$$\mathbf{D}_{ij} = \frac{k_b T}{6\pi\eta\sigma} \left[\left(1 - \frac{9 b_{ij}}{32\sigma} \right) \mathbf{I} + \frac{3 \mathbf{r}_{ij} : \mathbf{r}_{ij}}{\sigma b_{ij}} \right] \qquad i \neq j \tag{6}$$

for overlapping beads. In the previous example we have used the definitions $(\mathbf{r}:\mathbf{r})_{\alpha\beta} = \mathbf{r}_\alpha \mathbf{r}_\beta$, where α,β = 1, 2, 3 indicate the Cartesian coordinates x, y, z, and $b_{ij} = |\mathbf{r}_j - \mathbf{r}_i|$. The tensor in eq 6 is the analytical extension of eq 5 to the "unphysical" case of overlapping adjacent beads.[19]

The frictional and stochastic forces in eq 4 are connected via the relation:

$$\zeta_{ij} = \frac{1}{k_b T} \alpha_{ij} \cdot \alpha_{ij}^T \tag{7}$$

The Ermak–McCammon BD algorithm[18] is based on the assumption that the motions of interest occur on a time scale *much longer than the relaxation time of the velocity distribution* $\tau = m/\zeta$ (ζ = friction coefficient and m = mass of the monomer). In that case, integrating the Langevin equation twice, expanding the term $\tau_{ij} = \mathbf{D}_{ij} m/k_b T$ in a series of $\Delta\mathbf{r}_k(t) = \mathbf{r}_k(t) - \mathbf{r}_k(0)$, and truncation at the first order in $\Delta\mathbf{r}_k(t)$ leads to

$$\Delta\mathbf{r}_i(t) = \sum_{j=1}^{N} \left\{ \frac{\partial}{\partial \mathbf{r}_j} \mathbf{D}_{ij}(0) + \frac{1}{k_B T} \mathbf{D}_{ij}(0) \cdot \mathbf{F}_j(0) \right\} t + \mathcal{R}_i(t) = \mathbf{S}_i(0) t + \mathcal{R}_i(t) \tag{8}$$

$$\langle \mathcal{R}_i(t) : \mathcal{R}_j(t) \rangle = 2\mathbf{D}_{ij}(0) t \qquad \langle \mathcal{R}_i(t) \rangle = 0, \qquad i = 1, \ldots, N$$

This is the common Brownian dynamics algorithm used up to now.[11-14,18] We can write eq 8 in the more compact form

$$\underline{\mathbf{X}}(t) - \underline{\mathbf{X}}(0) = \underline{\mathbf{S}}(0) t + \underline{\mathcal{R}}(t)$$

$$\langle \underline{\mathcal{R}}(t) : \underline{\mathcal{R}}(t) \rangle = 2\underline{\mathbf{D}}(0) t \tag{9}$$

with the following definitions:

$$\{\underline{\mathbf{X}}(t)\}_j = \mathbf{r}_j(t) \quad \{\underline{\mathbf{S}}(t)\}_j = \mathbf{S}_j(t) \quad \{\underline{\mathcal{R}}(t)\}_j = \mathcal{R}_j(t)$$
$$\{\underline{\mathcal{R}}(t) : \underline{\mathcal{R}}(t)\}_{ij} = \mathcal{R}_i(t) : \mathcal{R}_j(t) \quad \{\underline{\mathbf{D}}(t)\}_{ij} = \mathbf{D}_{ij}(t) \tag{10}$$

For reviewing the second-order extension to the Ermak–McCammon algorithm (eq 8), we derive approximated solutions of the equation

$$\dot{\underline{\mathbf{X}}}(t) = \underline{\mathbf{S}}(t) + \underline{\mathcal{V}}(t) \tag{11}$$

where $\underline{\mathcal{V}}(t)$ are random velocities

$$\langle \underline{\mathcal{V}}(t) : \underline{\mathcal{V}}(t') \rangle = 2\delta(t - t')\underline{\mathbf{D}}(t)$$

As pointed out by Fixman,[20] eq 11 can be integrated by different approximations of the integral of $\underline{\mathbf{S}}(t)$ and

$\underline{\mathcal{V}}(t)$ over the interval $[0,t]$. The general solution is

$$\underline{\mathbf{X}}(t) = \underline{\mathbf{X}}(0) + t\underline{\mathcal{S}}(t) + \underline{\mathcal{R}}(t) \qquad \langle \underline{\mathcal{R}}(t):\underline{\mathcal{R}}(t)\rangle = 2t\underline{\mathcal{D}}(t) \tag{12}$$

with

$$\underline{\mathcal{S}}(t) = \frac{1}{t}\int_0^t \underline{\mathbf{S}}(z)\,dz \qquad \underline{\mathcal{D}}(t) = \frac{1}{t}\int_0^t \underline{\mathbf{D}}(z)\,dz \tag{13}$$

Integration by the Eulerian rule gives

$$\underline{\mathcal{S}}(t) = \underline{\mathbf{S}}(0) \qquad \underline{\mathcal{D}}(t) = \underline{\mathbf{D}}(0) \tag{14}$$

The solution of eq 11 in this approximation corresponds to the Ermak–McCammon algorithm (eq 8). On the other hand, if we integrate by the trapezoidal formula, we get

$$\underline{\mathcal{S}}(t) = (1/2)[\underline{\mathbf{S}}(t) + \underline{\mathbf{S}}(0)] \qquad \underline{\mathcal{D}}(t) = (1/2)[\underline{\mathbf{D}}(t) + \underline{\mathbf{D}}(0)] \tag{15}$$

This is the second-order BD algorithm proposed and applied by Iniesta and Garcia de la Torre for the trumbell dynamics.[15] In summary, we shall use two different algorithms to compute the trajectories of the molecule and compare the results obtained with the two. The first is given by

$$\mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + \mathbf{S}_i(t)\delta t + \mathcal{R}_i(t)$$

$$\langle \mathcal{R}_i(t):\mathcal{R}_j(t)\rangle = 2\mathbf{D}_{ij}(t)\delta t \qquad \langle \mathcal{R}_i(t)\rangle = 0 \qquad i = 1, ..., N \tag{16}$$

where δt is the simulation time step. We shall call eq 16 the first-order algorithm (FA). The second algorithm is given by

$$\mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + (1/2)[\underline{\mathbf{S}}_i(t+\delta t) + \underline{\mathbf{S}}_i(t)]\delta t + \underline{\mathcal{R}}_i(t)$$

$$\langle \underline{\mathcal{R}}_i(t):\underline{\mathcal{R}}_j(t)\rangle = [\underline{\mathbf{D}}_{ij}(t+\delta t) + \underline{\mathbf{D}}_{ij}(t)]\delta t$$
$$\langle \underline{\mathcal{R}}_i(t)\rangle = 0 \qquad i = 1, ..., N \tag{17}$$

where $\underline{\mathbf{S}}(t+\delta t)$ and $\underline{\mathbf{D}}(t+\delta t)$ have been calculated using the values of $\underline{\mathbf{X}}(t+\delta t)$ given by eq 16. Equation 17 will be called the second-order algorithm (SA).

We see that at each step the new conformation can be computed from the previous one by adding a deterministic and a stochastic term. For each time step δt, the SA formula requires approximately twice the number of computation steps as the FA formula.

b. Application of the Algorithm to Bead–Chain Models. We use here for the first time the SA formula with full hydrodynamic interaction on models of DNA fragments and compare the results with those obtained with the FA formula, using some conformational and dynamical parameters we shall describe in a later section.

The procedure is the following. We choose a number N of beads according to the size of the DNA fragment and select a starting conformation as outlined in appendix A. New conformations are computed using eqs 16 or 17. The time step δt must be chosen such that $\delta t \gg \tau$ ($\tau = m/\zeta$, the velocity relaxation time), and $\delta t \ll \tau_0$, where τ_0 is the maximum time for which the approximations 14 or 15 are valid. Obviously τ_0 will be different for the FA and SA algorithm.

The stochastic term $\underline{\mathcal{R}}(t)$, which has to conform to eq 16 or eq 17, is generated numerically using the procedure referred to by Ermak and McCammon.[18] This algorithm generates a set of nonindependent Gaussian-distributed random variables through a linear transformation of a set of independent Gaussian variables.

All programs were run on a Stardent 3000 computer and coded in double precision Fortran, using the compiler's vectorization option.

c. Tests of the Accuracy of the Numerical Simulations. To compare the FA and the SA formulas, we compute several conformational and dynamical molecular properties on the trajectories obtained from the two algorithms. One of them is the rms end-to-end distance S, defined as

$$\langle S^2\rangle = |\mathbf{r}_N - \mathbf{r}_1|^2 \tag{18}$$

We can compute S at each step of the simulation and compare the result with the one predicted by the Porod formula:

$$\langle S^2\rangle = 2LP\left\{1 - \frac{P}{L} + \frac{P}{L}e^{-P/L}\right\} \tag{19}$$

where L and P are the contour length and the persistence length of the DNA. We can also compute P from the simulation, using

$$P_f = b_o\sum_{j=1}^{N-2}\langle\cos(\alpha_j)\rangle = b_o\frac{1 - \langle\cos(\alpha)\rangle^{N-1}}{1 - \langle\cos(\alpha)\rangle} \tag{20}$$

This reduces to the known limit for P as $N \to \infty$:

$$P_\infty = \frac{b_o}{1 - \langle\cos(\alpha)\rangle} \tag{21}$$

where $\langle\cos(\alpha)\rangle = \langle\cos(\alpha_1)\rangle$, assuming a constant bending potential.

Since we allow for bond length variations by using a finite stretching potential between beads, eq 20 must be modified to give

$$P_s = \sum_{j=1}^{N-2}\frac{\langle b_{j+1}\cdot b_j\rangle}{\langle|b_j|\rangle} \tag{22}$$

In the case $\langle|b_j|\rangle = b_0$, i.e. an infinitely hard stretching potential, we get the usual definition (eq 20) for P. We shall discuss below the difference between eq 20 and eq 22 depending on the choice of δ.

The theoretical value of P, assuming fixed bond angles, can be computed from γ (eq 2) via $\langle\cos(\alpha)\rangle$:

$$\langle\cos(\alpha)\rangle = \frac{\int_0^\pi \cos(\alpha)e^{-U_b(\alpha)}\,d\alpha}{\int_0^\pi e^{-U_b(\alpha)}\,d\alpha} \tag{23}$$

In the case $\gamma \ll 1$, expanding the exponential in a Taylor series yields

$$\langle\cos(\alpha)\rangle = 1 - \gamma^2 + O(\gamma^4) \tag{24}$$

From $\langle\cos(\alpha)\rangle$ we can compute P_f (eq 20), or P_∞ (eq 21).

For the dynamical properties we focus on the translational diffusion coefficient D_t, which again may be computed in several different ways. First, D_t can be evaluated from displacement of the center of mass of the molecule $\mathbf{R}_{cm}$ using the Einstein relationship

$$\langle|\mathbf{R}_{cm}(t) - \mathbf{R}_{cm}(0)|^2\rangle = 6D_t t \tag{25}$$

Alternatively, we can compute D_t from the dynamic structure factor $S(\mathbf{K},t)$:

$$S(\mathbf{K},t) = \frac{1}{N^2}\left\langle\sum_{j,k=1}^{N} e^{-i\mathbf{K}(\mathbf{r}_j(t') - \mathbf{r}_k(t'+t))}\right\rangle_{t'} \tag{26}$$

where $\mathbf{K}$ is the scattering vector. At $|\mathbf{K}|R_g \ll 1$, R_g being

Table I
D_{cm} (eq 25), $\langle r \rangle$, and $\langle r^2 \rangle$ for the FA (I) and the SA (II) Algorithm[a]

δt	D_{cm}^{I}	D_{cm}^{II}	$\langle r \rangle^{I}$	$\langle r \rangle^{II}$	$\langle r^2 \rangle^{I}$	$\langle r^2 \rangle^{II}$
0.0003 (21 ps)	0.745	0.749	1.004	1.005	1.011	1.012
0.0009 (64 ps)	0.743	0.753	1.003	1.004	1.010	1.012
0.003 (210 ps)	0.73	0.76	1.003	1.003	1.005	1.009
0.0003 (21 ps)	0.744[b]					

[a] The theoretical values are $D_t = 0.75$, $\langle r \rangle = 1.005$, $\langle r^2 \rangle = 1.0125$;[18] δt is given in units $ut = 71$ (ns); $\sigma/d = 0.5$, $\delta = 0.05$. The 1σ error on D_{cm} is 1%, on $\langle r \rangle$ and $\langle r^2 \rangle$ 0.5%. [b] From the paper by Ermak and McCammon.[18]

the radius of gyration, the slope of ln $S(\mathbf{K},t)$ versus t is $-D_t\mathbf{K}^2$. The calculation of $S(\mathbf{K},t)$ can be greatly accelerated by using the relationship

$$S(\mathbf{K},t) = \langle \phi^*(\mathbf{K},t')\phi(\mathbf{K},t'+t) \rangle_{t'} \tag{27}$$

with

$$\phi(\mathbf{K},t) = \frac{1}{N}\sum_{j=1}^{N} e^{-i\mathbf{K}r_j(t)}$$

A third estimate of D_t can be obtained from the formula developed by de Haen et al.[21] on each configuration of the trajectory. We compute the average and the variance of this rigid body D_t on each trajectory and compare it with the values given by eqs 25 and 27.

d. Choice of the Time Step and Stretching Potential. We must respect the limits of validity of eqs 16 and 17. The time step δt of the simulation has to be larger than the velocity relaxation time τ, but must not be too large to make the approximated integration of eq 11 meaningless. A rough estimate of τ can be 5 ps, assuming a DNA density of 2. The upper limit of δt will be assessed from the quality of the simulations by computing the quantities defined in the preceding section and comparing them with the theoretical values.

Results

Iniesta and Garcia de la Torre[15] applied the second-order algorithm for the first time to BD simulations of simple models, such as dimers and trumbells. They compared the simulated tumbling rotational relaxation time for a trumbell with its theoretical value by computing this quantity from the slope of a plot of ln $(P_2(\mathbf{u}(t)\cdot\mathbf{u}(0))$ versus t, where $\mathbf{u}(t)$ is the end-to-end unit vector. We want to extend the simulations to longer chains. Since there is no analytical result for fragments with more than three beads, we have to do some cross checks between the different quantities defined in the previous section in order to estimate the quality of the simulation.

1. Dimers. The potential for a two-bead dimer is simply the stretching potential. We limit our analysis to the case of touching beads ($\sigma/d = 0.5$) since here the hydrodynamic interactions are strongest. The beads may overlap at some steps due to the finite stretching constant; in this case the analytical extension of the Rotne–Prager tensor (eq 6) is used. The values of the average bead-to-bead distance $\langle r \rangle$, the average squared distance $\langle r^2 \rangle$, and the translational diffusion coefficient D_t^{cm} from eq 25 are summarized in Table I. We used 100 000 time steps of δt for the FA, and 50 000 steps of $2\delta t$ for the SA formula. This way we get the same maximum simulation time and approximately also the same CPU time. We use normalized units for length, time, and diffusion coefficient: the unit of length (ul), time (ut), and diffusion coefficient (uD) are defined as d, d^2/D_0, and D_0, where D_0 is the diffusion coefficient of one single bead. In our choice $D_0 = 1.42 \times 10^{-10}$ (m^2 s^{-1}), $ut = 71.3$ (ns), and $ul = 3.184$ (nm).

Table II
Results of SA Simulations on a 10-Beads Fragment with $\gamma = 0.2523$ ($P_\infty = 50$ nm) and $\sigma/d = 0.5$[a]

δ	δt	$\langle r \rangle$	$\langle r^2 \rangle$	P_s, nm	P_∞, nm	P_f, nm	S, nm
0.07	0.002 [142 ps]	1.0093 (±0.0001)	1.025 (±0.0001)	22.7 (±0.2)	49 (±0.5)	22.2 (±0.1)	26.4 (±1)
0.03	0.0002 [14.2 ps]	1.0018 (±0.0002)	1.0045 (±0.0002)	24.4 (±0.3)	51 (±0.6)	22.5 (±0.2)	26.9 (±0.2)
0.03	0.0005 [35.6 ps]	1.0018 (±0.0002)	1.0049 (±0.0003)	22.9 (±0.2)	53 (±0.5)	22.6 (±0.2)	26.5 (±0.3)
0.07	0.004 [284 ps]	1.0074 (±0.0003)	1.023 (±0.0003)	22.6 (±0.3)	45 (±0.6)	21.7 (±0.3)	25.0 (±0.3)

[a] The theoretical values for P and S are $P_f = 24.1$ nm, $S = 28.8$ nm. The theoretical values for $\langle r \rangle$ and $\langle r^2 \rangle$ are $\langle r \rangle = 1.0098$, $\langle r^2 \rangle = 1.0245$, for $\delta = 0.07$, and $\langle r \rangle = 1.0018$, $\langle r^2 \rangle = 1.0045$ for $\delta = 0.03$. The values in parentheses are the errors (1σ), while those in square brackets are the δt values in picoseconds.

The results show the better precision obtained with the second-order algorithm. For the same accuracy in D_{cm} and the same CPU time, the total simulation time obtained is 3 times longer (data not shown).

2. Geometry of 10-Bead Chains. The optimum values for the stretching constant δ and the time step δt were chosen from SA simulations on a 10-bead fragment. We used $\sigma/d = 0.5$ as before, and $\gamma = 0.2523$, corresponding to $P_\infty = 50$ nm (eqs 21 and 24). The results are given in Table II.

The fluctuation of the average bead-to-bead distance is 7% for $\delta = 0.07$ and 3% for $\delta = 0.03$, as expected. The best agreement between the theoretical and the simulated $\langle r \rangle$ and $\langle r^2 \rangle$ values is obtained for $\delta t = 0.002$ ($\delta = 0.07$) and $\delta t = 0.0002$ ($\delta = 0.03$). P_∞ is computed within 2%, while P_f, P_s, and S are affected by larger errors (5%). In all simulations, P_s is in better agreement with the theoretical value ($P_f^{th} = 24.1$ nm) than P_f, thus confirming the choice of eq 22 over eq 20 for calculating the persistence length. From the results shown in Table II we chose a stretching constant $\delta = 0.07$ and a time step $\delta t = 0.002$ ($\delta t = 142$ ps for $\sigma/d = 0.5$) in all simulations reported in the following sections. With this choice of δt the average variation of the forces on a time step may be as high as 50–60%. However, as shown by the results, the simulation is still very stable. A slight improvement can be seen when going from $\delta = 0.07$ to $\delta = 0.03$. Thus, we believe that the errors which affect P_f, P_s, and S are due to the finite stretching potential. We cannot check this hypothesis by stiffening the stretching potential further using an even smaller δ, because this would imply a lower δt, violating the condition $\delta t \gg \tau = 5$ ps.

3. Diffusion Coefficient of Free-Draining Chains. D_t cannot be computed analytically for a number of beads $N > 3$, unless we assume a ratio $\sigma/d \ll 1$, where the hydrodynamic interactions can be neglected and $D_t^{th} = D_0/N$. Therefore we tested the accuracy of the SA formula on longer chains using $\sigma/d = 0.005$.

D_t was computed on five trajectories in three different ways. D_t^{cm} is obtained from the center-of-mass displacement (eq 25), D_t^{dif} is averaged over individual conformations of the trajectory using the modified Kirkwood equation by de Haen et al.,[21] and D_t^{sf} is the slope of ln $(S(\mathbf{K},t))/\mathbf{K}^2$ for $|\mathbf{K}|Nd = 0.1$. At $\mathbf{K}$ values such that $\mathbf{K}R_g \ll 1$, where R_g is the radius of gyration of the polymer, we expect $S(\mathbf{K},t)$ to be a single exponential decay with relaxation rate $\Gamma = D_t\mathbf{K}^2$. Our results show that at

Table III
Translational Diffusion Coefficient D_t^{cm} (eq 20), D_t^{dif} (after de Haen et al.[21]), and D_t^{sf} (eq 22 with $KL = 0.1$), Computed for $\sigma/d = 0.005$ with 20 000 Steps of the SA Algorithm (the error on the values is ±3% (1σ))

N	D_t^{cm}	D_t^{dif}	D_t^{sf}	D_t^{th}
10	0.095	0.102	0.093	0.1
17	0.057	0.060	0.065	0.059

Table IV
Translational Diffusion Coefficient D_t^{cm} (eq 20), D_t^{dif} (after de Haen et al.[21]), and D_t^{sf} (eq 22 with $KL = 0.1$), Computed on a 10-Subunit Chain for $\sigma/d = 0.5$ with 40 000 Steps of the SA or 80 000 Steps of the FA Algorithm (the error on the values is ±4% (1σ))

algorithm	D_t^{cm}	D_t^{dif}	D_t^{sf}
FA	0.283	0.284	0.277
SA	0.283	0.283	0.288

$|\mathbf{K}|Nd = |\mathbf{K}|L = 0.1$ (L = contour length) $S(\mathbf{K},t)$ does decay as a single exponential. At higher $\mathbf{K}$ in fact, $S(\mathbf{K},t)$ contains information on both the center of mass motion and the internal motions of the polymer.

Table III summarizes the results of the simulations with $N = 10$ and $N = 17$. The agreement between the simulated and theoretical values is within 1–2% for D_t^{dif}, and 5–7% for D_t^{cm} and D_t^{sf}.

4. Diffusion Coefficient of 10-Bead Chains with Hydrodynamic Interactions. For flexible chains with hydrodynamic interactions, we have performed simulations on a decamer ($N = 10$) with ratio $\sigma/d = 0.5$ and N_{max} = 40 000 for the SA and 80 000 for the FA equation. Since we cannot compute the exact value of D_t, we estimate the quality of the simulation by comparing the values of D_t^{dif}, D_t^{cm}, and D_t^{sf}. Table IV summarizes the results.

It is seen that D_t^{dif} and D_t^{cm} are very close for both the FA and SA equations, while the first cumulant diffusion coefficient shows the greatest variation.

5. Translational Diffusion Coefficient (D_t) and Rotational Relaxation Time (τ_r) from the BD Simulation Trajectory. One of our aims is to simulate the hydrodynamic parameters of flexible DNA filaments as measured in a DLS experiment. In order to check the precision of such a simulation, we computed the dynamic structure factor $S(\mathbf{K},t)$ of a dimer with negligible hydrodynamic interactions ($\sigma/d = 0.005$), averaging over seven trajectories of $N = 40\,000$ steps each. The first cumulant diffusion coefficient D_{cum}, obtained from the slope of ln $(S(\mathbf{K},t))/\mathbf{K}^2$ (Figure 1), increases from 0.5 to 0.9 as $\mathbf{K}$ is increased. Theoretically, at low $\mathbf{K}$ one should observe the center-of-mass motion of the dimer ($D_t = 0.5$ without hydrodynamic interaction), while in the high $\mathbf{K}$ limit only the monomer motion should be seen ($D_t = 1$). However, as is clearly evident from the plot, D_{cum} does not reach 1.0; furthermore, it starts oscillating for $\mathbf{K}^2d^2 > 10.0$ nu. The reason for this behavior, as has already been pointed out by Allison et al.,[13] is the presence of a finite stretching potential. In this aspect, the SA algorithm behaves exactly like the FA. Adding hydrodynamic interactions (σ/d = 0.5) smoothes out the oscillations, and although the plateau of the D_{cum} versus $\mathbf{K}^2$ curve does not quite reach 1.0, it comes very close to the theoretical value. At present we do not have a good explanation for this behavior, and as a consequence one should confine the interpretation of the results to data below $\mathbf{K}^2d^2 = 10$. This does not really limit the applicability of the model to real-world problems; for example, in the case of DNA, the maximum usable $\mathbf{K}$ $\approx 10^{-9}$ m^{-1}, a value only reached in spin–echo neutron scattering experiments.[22]

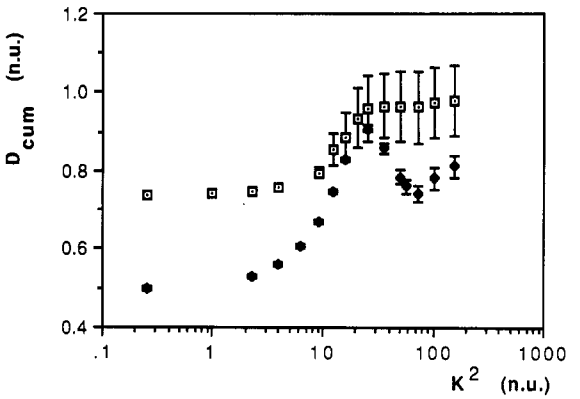

Figure 1. First cumulant diffusion coefficient D_{cum} for a dimer in the case of high ($\sigma/d = 0.5$; □) and negligible ($\sigma/d = 0.005$; ●) hydrodynamic interactions, plotted versus the square of the scattering vector $\mathbf{K}$. All the quantities are expressed in normalized units (see text).

Table V
End-to-End Distances $\langle r \rangle$, First Cumulant Translational Diffusion Coefficients D_t, and Rotational Relaxation Rates (see text) of a Dimer with Hydrodynamic Interactions (errors given are 1σ)

$\langle r \rangle^I$	$\langle r \rangle^{II}$	D_t^I	D_t^{II}	Γ_1^I	Γ_1^{II}
1.0096 (±0.0005)	1.0098 (±0.0004)	0.52 (±0.03)	0.508 (±0.01)	3.95 (±0.04)	4.00 (±0.04)

Table VI
Dimer with $\sigma/d = 0.005$ and $N_{max} = 40\,000$[a]

DLS		FPA	
$(\tau_r)^I$	$(\tau_r)^{II}$	$(\tau_2)^I$	$(\tau_2)^{II}$
46.0 ps	59.72 ps	60.2 ps	59.4 ps

[a] For definitions of τ_r and τ_2, see the text. Theoretical value of $\tau_r = \tau_2 = 59.6$ ps. The error on the quantities (1σ) is ±5%.

An extrapolation of D_{cum} versus $\mathbf{K}^2 \to 0$ yields the translational diffusion coefficient D_t. The slope Γ_1 of ln $\langle P_1(\mathbf{u}(t)\cdot\mathbf{u}(0))\rangle$ versus t is equal to $(3\tau_r)^{-1}$, where τ_r is the end-over-end rotational relaxation time of the dimer.[23] Table V summarizes the results for the SA and FA case, together with the mean end-to-end distance $\langle r \rangle$.

The SA results again are in better agreement with the theoretical values, which are $\langle r \rangle = 1.0098$, $D_t = 0.5025$, and $\Gamma_1 = 3.985$.[18]

6. Multiexponential Behavior of the Dynamic Structure Factor. For further testing the applicability of the SA and FA formulas to the simulation of DLS decays, we computed the structure factor $S(\mathbf{K},t)$ at $KL = 6.0$, and the average of the second Legendre polynomial $P_2(\mathbf{u}(t)\cdot\mathbf{u}(0))$. This last quantity decays single exponentially with a relaxation time equal to the rotational time τ_r of the dimer and is measured, for example, in a fluorescence polarization anisotropy decay (FPA) experiment.[6] The maximum simulation time was 1.1 μs, and $S(\mathbf{K},t)$ and $P_2(\mathbf{u}(t)\cdot\mathbf{u}(0))$ were computed up to $t_{DLS} = 1.5$ ns, while the relaxation time for translational diffusion was 0.16 ns. For the DLS simulation, the translational diffusion coefficient D_t was held fixed at its theoretical value of 0.5025, and a double-exponential decay with relaxation rates $\Gamma_1 = \mathbf{K}^2D_t$ and $\Gamma_2 = \mathbf{K}^2D_t + 1/\tau_r$ was fitted to $S(\mathbf{K},t)$ by a nonlinear least-squares Marquardt algorithm. Table VI shows the τ_r values obtained in this way for the FA and SA models. The theoretical value is $\tau_r = \tau_2 = 59.6$ ps, very well predicted by the SA model. τ_2 calculated from the DLS decay for the FA simulation shows a 23% deviation from the theoretically predicted value.

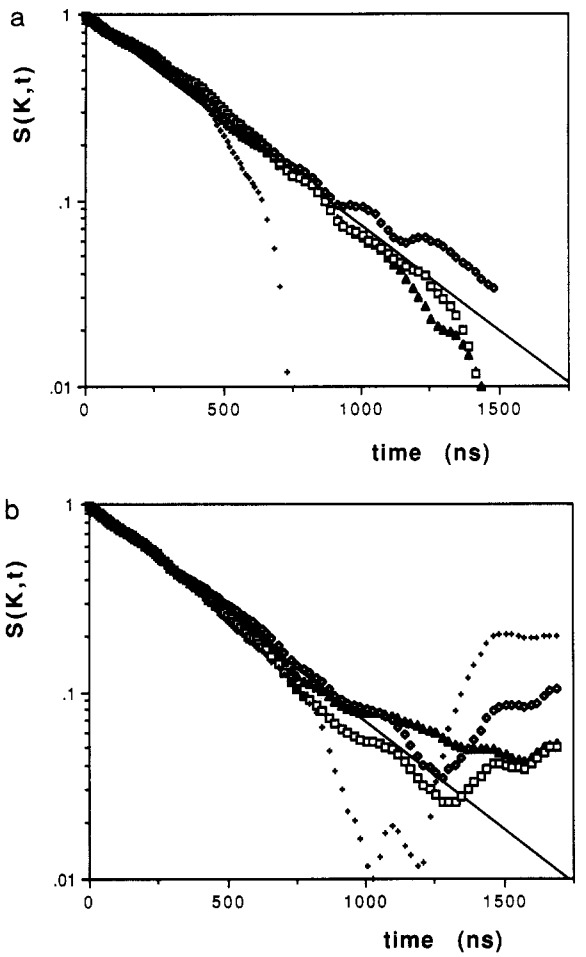

Figure 2. The dynamic structure factor decay versus time (ns) for a dimer at $\mathbf{K}$ = 1.0 nu and for the SA algorithm (a) and for the FA algorithm (b). The symbols refer to trajectories of different length: (▲) t_{max} = 11 μs, (□) t_{max} = 9 μs, (◇) t_{max} = 6 μs, and (+) t_{max} = 3 μs.

7. Minimum Length of the Simulation. To get a good approximation to the full DLS decay, $S(\mathbf{K},t)$ should be computed to a maximum time t_{DLS} of at least 4–5 mean relaxation times τ. Furthermore, the computation should extend over a reasonably long trajectory in order to get a good average. The total time t_{max} necessary to get a good estimate of $S(\mathbf{K},t)$ for a given t_{DLS} was checked on a dimer model with hydrodynamic interactions: d = 3.184 nm and σ = 1.592 nm. We computed $S(\mathbf{K},t)$ on trajectories of 40 000, 30 000, 20 000, and 10 000 steps for $|\mathbf{K}|L$ = 1.0, always using the same step length δt = 2.85 ns, which gives t_{max} values of 11.4, 8.5, 5.7, and 2.85 μs for the four cases. The theoretical τ value for this simulation is 0.38 μs, thus $t_{DLS} = 4\tau$ = 1.4 μs. A single-exponential fit to the first 40 ns of $S(\mathbf{K},t)$ yields t = 0.386 μs, in very good agreement with the theoretical value. From the plot of $S(\mathbf{K},t)$ versus t (Figure 2) we see that satisfactory results are obtained with $t_{max} = 10t_{DLS}$. Since we generally find that using several short trajectories starting from different random configurations gives better results than using one long trajectory, this is also a practical upper limit for the length of the simulation.

Conclusion

We have used the second-order Brownian dynamics algorithm proposed by Iniesta and Garcia de la Torre[15] to calculate the hydrodynamic properties of various bead–chain models of linear polymers. The parameters of the model—hydrodynamic chain diameter and bending rigidity—were chosen equal to the known values for DNA. The constraint of constant length was imposed by introducing a stiff stretching potential. A torsional potential was not included in this version of our model; the quantities computed here do not depend on the torsional dynamics. The diffusion coefficient only depends on the center-of-mass motion, and it has been shown that torsional and bending deformations are statistically independent for a chain with an isotropic bending potential.[17] Furthermore, the motions we are interested in occur on the time scale of bending deformations.[24]

We compared the algorithm with the results given by the widely used first-order BD procedure.[18] The improvement of the second-order algorithm over the first-order procedure is such that in general 4–5 times less CPU time may be used to produce the same precision in the prediction of the translational diffusion coefficient for models consisting of 2–17 beads, with and without hydrodynamic interaction. If we try to extract the rotational relaxation time in addition to the translational diffusion coefficient from simulated DLS auto-correlation functions by a biexponential fit, we see an even more significant improvement for the second-order procedure; for an equal number of simulation time steps, the second-order result matches the theoretical value within 0.2%, while the first-order algorithm deviates by 23%. This deviation cannot be explained by large uncertainties in the simulations, because the statistical error (6 trajectories of 40 000 steps each) was only 5% (1σ). We assume that the deviation for the FA procedure reflects a systematic error inherent to the simulation algorithm, which shows up here because extracting a second exponential decay from noisy data is particularly sensitive to any such errors. If one calculates the rotational relaxation time from the same trajectories through a more direct route, such as the simulated fluorescence polarization anisotropy decay, where only a single exponential fit is required, the difference between the two procedures becomes much less pronounced.

The algorithm described here is being extended to accommodate a torsion potential between the beads, and in that way may be used to compute the dynamical properties of circular and supercoiled DNA chains, as well as DNAs with permanently curved sequences. In the latter case, a torsional potential must be included even for linear DNA chains, because the curve creates a torsional correlation between distant parts of the chain. These studies are presently underway.

Appendix A

The initial positions of the N beads $\{\underline{\mathbf{r}}_j\}$ are computed from the values $\{b_j\}$, $\{\alpha_j\}$, and $\{\phi_j\}$ through the relations

$$\underline{\mathbf{r}}_1 = (0, 0, 0) \qquad \underline{\mathbf{r}}_2 = (0, 0, b_1)$$

$$\underline{\mathbf{r}}_{j+2} = \underline{\mathbf{r}}_{j+1} + b_{j+1}\,[\cos(\phi_j)\sin(\alpha_j), \sin(\phi_j)\sin(\alpha_j), \cos(\alpha_j)] \quad \text{(A1)}$$

The distribution of b_j is $P(b_j) = \exp\{-(b_j^2 - b_0^2)/(2\delta^2)\}$, while that of α_j is $\sin(\alpha_j)P(\alpha_j)$, where $P(\alpha_j) = \exp\{-(\alpha_j^2)/(2\gamma^2)\}$. We have used a standard routine (GASDEV) for a Gaussian distribution,[25] to generate $P(b_j)$ and $P(\alpha_j)$, and generated the distribution $\sin(\alpha_j)P(\alpha_j)$ taking a number $0 < \alpha < \pi$ belonging to the distribution $P(\alpha_j)$ and comparing $\sin(\alpha_j)$ with a random number $0 < x < 1$. If $\sin(\alpha_j) > x$, then α is taken as a number belonging to the distribution $\sin(\alpha_j)P(\alpha_j)$. The numbers $\{\phi_j\}$ are randomly generated in the $[0,2\pi]$ interval.

References and Notes

(1) Langowski, J.; Kremer, W.; Kapp, U. Dynamic light scattering for the study of the solution conformation and dynamics of superhelical DNA. *Methods in Enzymology*; Lilley, D., Dahlberg, J., Eds.; Academic: London, in press; Vol. DNA Structure.
(2) Booles, T. C.; White, J. H.; Cozzarelli, N. R. *J. Mol. Biol.* **1990**, *213*, 931.
(3) White, J. H.; Bauer, W. R. *J. Mol. Biol.* **1986**, *189*, 329.
(4) Langowski, J.; Giesen, U.; Lehmann, C. *Biophys. Chem.* **1986**, *25*, 191.
(5) Chirico, G.; Baldini, G. *J. Mol. Liq.* **1989**, *41*, 327.
(6) Berne, B. J.; Pecora, R. *Dynamic Light Scattering*; J. Wiley and Sons: New York, 1976.
(7) Pecora, R. *J. Chem. Phys.* **1964**, *40*, 1604.
(8) Lin, S. C.; Schurr, J. M. *Biopolymers* **1978**, *17*, 425.
(9) Soda, K. *Macromolecules* **1984**, *17*, 2365.
(10) Berg, O. G. *Biopolymers* **1979**, *18*, 2861.
(11) Allison, S. A.; McCammon, J. A. *Biopolymers* **1984**, *23*, 363.
(12) Allison, S. A. *Macromolecules* **1986**, *19*, 118.
(13) Allison, S. A.; Sorlie, S. S.; Pecora, R. *Macromolecules* **1990**, *23*, 1110.
(14) Lewis, R. G.; Allison, S. A.; Eden, D.; Pecora, R. *J. Chem. Phys.* **1988**, *89* (4), 2490.
(15) Iniesta, A.; Garcia de la Torre, J. *J. Chem. Phys.* **1990**, *92* (3), 2015.
(16) Hagerman, P. J.; Zimm, B. H. *Biopolymers* **1981**, *20*, 1481.
(17) Schurr, J. M. *Biopolymers* **1985**, *24*, 1233.
(18) Ermak, D. L.; McCammon, J. A. *J. Chem. Phys.* **1978**, *69* (4), 1352.
(19) Rotne, J.; Prager, S. *J. Chem. Phys.* **1969**, *50* (11), 4831.
(20) Fixman, M. *Macromolecules* **1986**, *19*, 1195.
(21) de Haen, C.; Easterly, R. A.; Teller, D. C. *Biopolymers* **1983**, *22*, 1133.
(22) Neutron Spin Echo. *Lecture Notes in Physics*; Mezei, F., Ed.; Springer: New York, 1980; Vol. 128.
(23) Diaz, F. G.; Garcia de la Torre, J. *J. Chem. Phys.* **1988**, *88* (12), 7698.
(24) Langowski, J.; Fujimoto, B. S.; Wemmer, D.; Benight, A. S.; Drobny, G.; Shibata, J. H.; Schurr, J. M. *Biopolymers* **1985**, *24*, 1023.
(25) Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; Vetterling, W. T. *Numerical Recipes*; Cambridge University Press: Cambridge, 1986.
(26) Allison, S.; Austin, R.; Hogan, M. *J. Chem. Phys.* **1989**, *90* (7), 3843.