Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2008) 2794–2807
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

An effective bead–spring model for polymer simulation

Bin Liu [a,*], Jizeng Wang [b,*], Xiaojun Fan [b], Yong Kong [c], Huajian Gao [b]

[a] FML, Department of Engineering Mechanics, Tsinghua University, Beijing 100084, China
[b] Division of Engineering, Brown University, Providence, RI 02912, USA
[c] ICX Radiation GmbH, Koelner Str. 99, 42651 Solingen, Germany

Received 4 April 2007; received in revised form 15 October 2007; accepted 12 November 2007
Available online 21 November 2007

Abstract

An effective bead–spring model combining the advantages of large time steps of traditional bead–rod models and computational rigor of traditional bead–spring models is proposed to simulate the dynamic behaviors of flexible polymer chains with arbitrary longitudinal stiffness. The proposed model can be used to simulate many types of polymer chains or networks with different chain elasticity via a unified integration scheme with reasonably large time steps.

Keywords: Bead–spring model; Bead–rod model; Polymer dynamics; Brownian dynamic simulations; Flexible polymer; Freely jointed chain model

1. Introduction

For a long flexible polymer chain subject to a tensile stress [1,2], thermal fluctuations and undulations of the chain will be gradually frozen out as the stress increases so that, eventually, the deformation behavior of the chain will be dominated by its longitudinal elasticity instead of configurational entropy [3]. There are currently no unified polymer simulation models to treat the continuous transition from entropy dominated to bond-stretching dominated regimes. Depending on the longitudinal stiffness of a polymer chain, at least three types of coarse grained simulation models have been developed in the literature: the linear bead–spring model for chains with small longitudinal stiffness [4–7], the bead–nonlinear-spring model [8–11] and the bead–rod model [10–15] for chains with large longitudinal stiffness. In these methods, a continuous polymer chain is discretized as a series of virtual beads connected by springs or by rigid rods.

Among various discrete models for polymer chains, the linear bead–spring model is apparently suitable for the dynamic simulations of chains with longitudinal elasticity since neither the bead–nonlinear-spring model nor the bead–rod model can properly describe the backbone extension when the chains are significantly

* Corresponding authors. Tel.: +86 10 62786194; fax: +86 10 62781824 (B. Liu), tel.: +1 401 598 7392; fax: +1 401 863 1309 (J. Wang).
E-mail addresses: liubin@tsinghua.edu.cn (B. Liu), jizeng_wang@brown.edu (J. Wang).

0021-9991/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2007.11.012

stretched. However, in order to ensure numerical convergence, the time step in the simulations of the bead–spring chain must be taken less than the shortest intrinsic vibration period of the springs. This requirement makes the bead–spring type models less efficient in the case of short vibration periods under large longitudinal stiffness. In this case, the choice of simulation methods often has to go to a bead–rod type model which can take larger time steps because the high-frequency vibration modes are neglected by replacing the connecting springs with inextensible rods. The cost for such replacement is that the backbone extension of the chain is neglected while additional constraint conditions on the rod length must be enforced in each integration time step. Nonlinear spring models have also been introduced in order to describe the properties of a polymer chain which becomes increasingly stiff under stretch, examples including the inverse Langevin spring model [16] and the wormlike spring model [1]. A drawback of nonlinear spring models is that the spring forces can become singular when approaching full extension [9].

In this paper, we propose an effective bead–spring model which combines the merits of large time steps of bead–rod type models and computational rigor of the bead–spring type models. The model incorporates effective constraint forces in a second-order approximation and appears to be accurate in both entropy-dominated and bond-dominated regimes. Because of this wide range of applicability and efficiency, the proposed model can be used to simulate complex polymer systems via an efficient unified integration scheme with large integration time steps.

2. Model

For a polymer with low or intermediate longitudinal stiffness, the bead–spring type models are quite efficient since the following simple integration scheme can be adopted [17],

$$\mathbf{X}^{(n+1)} = \mathbf{X}^{(n)} + \frac{\Delta t}{k_{\mathrm{B}}T}\mathbf{D}^{(n)}\mathbf{F}^{(n)} + \mathbf{R}^{(n)} \tag{1}$$

where $\mathbf{X}^{(n+1)}$ and $\mathbf{X}^{(n)}$ are the new and current position vectors of a set of virtual beads, $\mathbf{D}^{(n)}$ is the translational diffusion matrix, and $\mathbf{F}^{(n)}$ is the vector of forces acting on the beads. The statistical random force (in the dimension of length), $\mathbf{R}^{(n)}$, which is generated at each time step according to a Gaussian distribution with zero mean and variation, satisfies

$$\langle \mathbf{R}^{(n)}\mathbf{R}^{(n')} \rangle = 2\mathbf{D}^{(n)}\Delta t\delta_{nn'} \tag{2}$$

where $\delta_{nn'}$ is the Kronecker delta.

To ensure numerical convergence, the time step Δt in the bead–spring model must be smaller than the shortest intrinsic vibration period of the springs even though the overall dynamic behaviors of the chains in entropy-dominated cases may not be significantly affected by the longitudinal elasticity [14]. This requirement makes the bead–spring type models less efficient in the case of large longitudinal stiffness. In this case, the choice of simulation methods often goes to a bead–rod type model which can take larger time steps because the high-frequency vibration modes are neglected by replacing the connecting springs with inextensible rods. The computational cost for this replacement is that an additional $N-1$ constraint conditions (N = number of beads in the polymer chain) on the rod length must be satisfied in each integration time step. An efficient integration scheme for the bead–rod model is the so-called linear constraint solver [18,14], in which the bead positions are updated according to

$$\mathbf{X}^{(n+1)} = (\mathbf{I} - \mathbf{T}^{(n)}\mathbf{B}^{(n)})\left(\mathbf{X}^{(n)} + \frac{\Delta t}{k_{\mathrm{B}}T}\mathbf{D}^{(n)}\mathbf{F}^{(n)} + \mathbf{R}^{(n)}\right) + \mathbf{T}^{(n)}\mathbf{d} \tag{3}$$

where $\mathbf{d}$ is the vector of $(N-1)$ inextensible rod length, $\mathbf{B}^{(n)}$ is an $(N-1)\times 3N$ matrix containing the gradients of the constraints, and

$$\mathbf{T}^{(n)} = \mathbf{D}^{(n)}\mathbf{B}^{(n)^T}(\mathbf{B}^{(n)}\mathbf{D}^{(n)}\mathbf{B}^{(n)^T})^{-1} \tag{4}$$

Since the computation of $\mathbf{T}^{(n)}$ involves matrix inversion, the computational effort for each time step in the bead–rod model is usually much larger than that in the bead–spring approach. In the case of chain networks, the issue of computational cost can be even more serious due to the lack of effective treatments or approximations for

simplifying the matrix inversion [18,14]. Although various nonlinear spring models [8–11] have also been developed for polymers with inextensible backbones by limiting vibrations along the chain length, these models usually suffer the drawback of singular spring forces as the chain approaches full extension [9].

In this paper, we propose an effective bead–spring model to combine the merits of large time steps of the bead–rod type models and computational rigor of the bead–spring type models. There are usually two stages in simulating polymers: on the first stage, a real polymer molecule is simplified to an idealized model, such as a bead–spring chain; on the second stage, a proper integration algorithm is selected to simulate the behaviors of the ideal model. Here we focus on the second stage, i.e. developing an accurate and efficient integration algorithm. In our approach, the integration scheme for the bead–spring model is used but the spring constant is replaced with an effective constant in the first-order approximation and, for higher accuracy, additional effective constraint forces are introduced in the second-order approximation which appears to be accurate in both entropy-dominated and bond-dominated regimes.

We first demonstrate the basic idea with a simple example involving a pair of beads connected via a spring with stiffness constant k subject to an external force f, as shown in Fig. 1(a). With the typical assumption in Brownian dynamics that inertia effects are negligible, the governing equation for the motion of one bead (here, without losing generality, only the right half of the system is studied due to the symmetry) is

$$-\xi\dot{x} - 2kx + f = 0 \tag{5}$$

where the center of the spring is fixed, ξ is the friction coefficient and the origin of x corresponds to the equilibrium position of the bead. The solution to Eq. (5) shows that the bead position at time $t = \Delta t$ is

$$x(\Delta t) = x(0)\mathrm{e}^{-\frac{2k}{\xi}\Delta t} + \frac{f}{2k}\left(1 - \mathrm{e}^{-\frac{2k}{\xi}\Delta t}\right) \tag{6}$$

We can compare the above analytical solution to the integration scheme of the bead–spring type models in Eq. (1). For $D^{(n)} = k_{\mathrm{B}}T/\xi$ and

$$F^{(n)} = -2kx^{(n)} + f \tag{7}$$

Eq. (1) becomes

$$x^{(n+1)} = x^{(n)}\left(1 - \frac{2k\Delta t}{\xi}\right) + f\frac{\Delta t}{\xi} \tag{8}$$

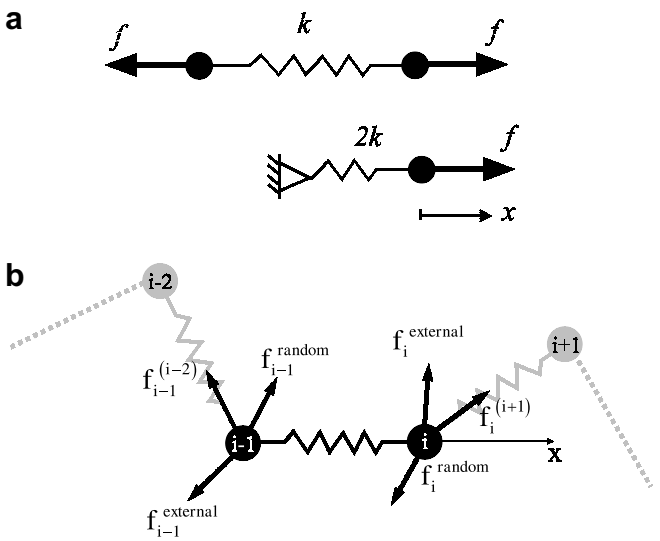

Fig. 1. An illustrative diagram of two beads subjected to forces. (a) A stand-alone pair of beads connected by a spring. (b) Free body diagram of a pair of beads from a bead–spring chain.

If the time step n and $n+1$ correspond to $t=0$ and $t=\Delta t$, respectively, matching the prediction of the bead position from the analytical solution in Eq. (6) and that from the integration scheme in Eq. (8) requires

$$1-\frac{2\Delta tk}{\xi}=\mathrm{e}^{-\frac{2\Delta tk}{\xi}} \tag{9a}$$

$$\frac{\Delta t}{\xi}=\frac{1-\mathrm{e}^{-\frac{2k}{\xi}\Delta t}}{2k} \tag{9b}$$

Eqs. (9a) and (9b) are approximately satisfied if $\Delta tk/\xi \ll 1$, which is just the requirement for time steps in the bead–spring type models: smaller time step Δt should be taken at larger stiffness k.

To overcome this restriction of small time step for the spring with high stiffness, as mentioned above, bead–rod type models may be used but the extensions of rods are always zero no matter how large the external force f is, which leads to certain sacrifice of computational rigor due to the approximation of replacing a stiff spring with an inextensible rod.

To keep the simplicity of the time integration scheme for bead–spring models, we propose an effective bead–spring model which can take large time steps even for chains with large longitudinal stiffnesses and can simulate the chain behaviors in both entropy-dominated and bond-stretching dominated regimes. Introducing two effective parameters in this effective bead–spring model, the effective spring constant $\tilde{k}$ and the effective constraint force $\tilde{f}$ (taking positive value when the force is inward), we modify Eq. (7) as

$$F^{(n)}=-2\tilde{k}x^{(n)}-\tilde{f}+f. \tag{10}$$

Substituting Eq. (10) into Eq. (1) yields the following revised integration scheme:

$$x^{(n+1)}=x^{(n)}\left(1-\frac{2\tilde{k}\Delta t}{\xi}\right)+(f-\tilde{f})\frac{\Delta t}{\xi}. \tag{11}$$

To accurately predict the bead position from Eq. (11), the effective spring constant $\tilde{k}$ and constraint force $\tilde{f}$ can be determined by comparing Eqs. (11) and (6) as

$$\tilde{k}=\frac{1}{2}\left(1-\mathrm{e}^{-\frac{2\Delta tk}{\xi}}\right)\frac{\xi}{\Delta t} \tag{12a}$$

$$\tilde{f}=\left[1-\frac{\xi}{2\Delta tk}\left(1-\mathrm{e}^{-\frac{2\Delta tk}{\xi}}\right)\right]f \tag{12b}$$

Obviously, the revised integration scheme in Eq. (11) can be used to simulate this simple system accurately for any time step Δt.

The proposed effective bead–spring approach can actually be applied to polymer chains with arbitrary longitudinal stiffness. One can see this clearly in the following extreme cases. In the limit when $2\Delta tk/\xi \ll 1$, Eqs. (12a) and (12b) become

$$\tilde{k}=\frac{1}{2}\left(1-\mathrm{e}^{-\frac{2\Delta tk}{\xi}}\right)\frac{\xi}{\Delta t}\approx\frac{1}{2}\left[1-\left(1-\frac{2\Delta tk}{\xi}\right)\right]\frac{\xi}{\Delta t}=k \tag{13a}$$

$$\tilde{f}=\left[1-\frac{\xi}{2\Delta tk}\left(1-\mathrm{e}^{-\frac{2\Delta tk}{\xi}}\right)\right]f\approx\left[1-\frac{\xi}{2\Delta tk}\left(\frac{2\Delta tk}{\xi}\right)\right]f=0 \tag{13b}$$

indicating that the effective model would degenerate into the traditional bead–spring model when the spring constant k or the time step Δt is sufficiently small. In the other limit when $k\to\infty$, Eqs. (13a) and (13b) are reduced to

$$\tilde{k}=\frac{1}{2}\frac{\xi}{\Delta t} \tag{14a}$$

$$\tilde{f}=f \tag{14b}$$

Substituting Eq. (14) into Eq. (11) yields

$$x^{(n+1)} = x^{(n)}\left[1 - \frac{2\left(\frac{\xi}{2\Delta t}\right)\Delta t}{\xi}\right] = 0 \tag{15}$$

corresponding to the rigid-rod constraint. Eq. (15) shows that we can use the bead–spring integration scheme in Eq. (11) to simulate the chain with infinite longitudinal stiffness as long as an effective spring constant and constraint force are taken as $\tilde{k} = \frac{\xi}{2\Delta t}$ and $\tilde{f} = f$, respectively.

It should be pointed out that the introduction of these effective parameters is just a numerical scheme to make the simulation independent of the choice of the time step (at least for large time step) and, as such, does not change real physical problem at all. The essence of this effective bead–spring model is to make use of an analytical solution to achieve large time steps in numerical simulations.

This effective treatment can be easily verified for the one spring system under constant forces, and we will demonstrate in the following that it can be extended to more general cases. Consider the free body diagram of a pair of beads connected by a spring from a bead–spring chain, as illustrated in Fig. 1(b). There are a total of six forces acting on this subsystem, which include the statistical random forces $\mathbf{f}_i^{\text{random}}, \mathbf{f}_{i-1}^{\text{random}}$, the external forces $\mathbf{f}_i^{\text{external}}, \mathbf{f}_{i-1}^{\text{external}}$ and neighboring spring forces $\mathbf{f}_{i-1}^{(i-2)}, \mathbf{f}_i^{(i+1)}$. The variation of the spring-length, characterized by the local coordinate x in Fig. 1(b), should only depend on the x-axis projection of the force difference between bead # i and bead # $i-1$, i.e., $[(\mathbf{f}_i^{\text{external}} + \mathbf{f}_i^{\text{random}} + \mathbf{f}_i^{(i+1)}) - (\mathbf{f}_{i-1}^{\text{external}} + \mathbf{f}_{i-1}^{\text{random}} + \mathbf{f}_{i-1}^{(i-2)})] \cdot \mathbf{e}_x$, where $\mathbf{e}_x$ is the unit vector along the x-axis. In comparison with the previous simple example, this is equivalent to taking the stretching force f as

$$f = \frac{1}{2}[(\mathbf{f}_i^{\text{external}} + \mathbf{f}_i^{\text{random}} + \mathbf{f}_i^{(i+1)}) - (\mathbf{f}_{i-1}^{\text{external}} + \mathbf{f}_{i-1}^{\text{random}} + \mathbf{f}_{i-1}^{(i-2)})] \cdot \mathbf{e}_x. \tag{16}$$

Different from the previous simple example, f in this complex system relies on many factors and can be viewed as a statistical variable with the mean value $\langle f \rangle$.

According to Eq. (12), $\tilde{k}$ depends on the spring constant k only, while the effective constraint forces $\tilde{f}$ also depends on f, and therefore it should be updated frequently to obtain the exact evolution of the system. However, considering that there exist random forces $\mathbf{f}_i^{\text{random}}$ and $\mathbf{f}_{i-1}^{\text{random}}$ in Eq. (16), this process is not deterministic but statistical. In order to save the computation effort while capturing the average behaviors of the chain, $\tilde{f}$ can be related to $\langle f \rangle$ instead of f as

$$\tilde{f} = \left[1 - \frac{\xi}{2\Delta t k}\left(1 - \mathrm{e}^{-\frac{2\Delta t k}{\xi}}\right)\right]\langle f \rangle \tag{17}$$

Moreover, it is found that, for a specific $x^{(n)}$ or $x(0)$, the mean value $\langle x^{(n+1)} \rangle$ from Eq. (11) and $\langle x(\Delta t) \rangle$ from Eq. (6) are identical, i.e. the average spring extension can be accurately predicted with this simple effective bead–spring model.

It should be emphasized that, for bead–spring system with identical springs, the parameter $\tilde{k}$ in Eq. (12a), which depends on the spring constant k, is a constant for identical springs, while the effective constraint forces $\tilde{f}$ in Eq. (17) may take different values for different springs even with the same stiffness k since $\langle f \rangle$ is not a constant. In general cases, different springs need different $\tilde{k}$ and $\tilde{f}$. The integration scheme for the whole system in Eq. (1) then becomes

$$\mathbf{X}^{(n+1)} = \mathbf{X}^{(n)} + \frac{\Delta t}{k_{\mathrm{B}}T}\mathbf{D}^{(n)}\widetilde{\mathbf{F}}^{(n)} + \mathbf{R}^{(n)} \tag{18}$$

where $\widetilde{\mathbf{F}}^{(n)}$ is the collective vector of effective forces acting on the beads, i.e. the sum of real external forces, effective spring forces and effective constraint forces. It can be similarly reasoned from the derivation above that the proposed integration scheme is also applicable for polymer networks without additional computational complexity and cost.

When the time step Δt is chosen, the effective spring constants $\tilde{k}$ can be computed from Eq. (12a). However, the effective constraint forces $\tilde{f}$ in Eq. (17) cannot be determined since we do not know $\langle f \rangle$ before the simulation. We then start the simulation with an initial guess of $\tilde{f}^{\text{guessed}} = 0$, and call it the first-order effective bead–spring model, i.e. a scheme with only effective spring constants $\tilde{k}$. After a round of simulation, $\langle f \rangle$ can be computed from the information collected. Specifically, averaging Eq. (11) yields

$$\langle x^{(n+1)}\rangle = \langle x^{(n)}\rangle\left(1-\frac{2\tilde{k}\Delta t}{\xi}\right)+(\langle f\rangle-\tilde{f}^{\text{guessed}})\frac{\Delta t}{\xi} \tag{19}$$

Considering $\langle x^{(n+1)}\rangle = \langle x^{(n)}\rangle = \langle x\rangle$, $\langle f\rangle$ can be solved from the equation above as

$$\langle f\rangle = 2\tilde{k}\langle x\rangle + \tilde{f}^{\text{guessed}} \tag{20}$$

Once the estimation of $\langle f\rangle$ is obtained, the $\tilde{f}^{\text{guessed}}$ can be updated according to Eq. (17). It will be shown later that, just one-time updating of $\tilde{f}^{\text{guessed}}$ is able to yield quite accurate results in most situations. For systems evolving statistically with time, regular updating of $\tilde{f}^{\text{guessed}}$ may be required. We call the above integration scheme with both effective spring constants $\tilde{k}$ and non-zero effective constraint forces $\tilde{f}$ the second-order effective bead–spring model.

In the effective bead–spring model, the time step Δt is no longer constrained by the convergence conditions due to high or infinite spring stiffness, but we still need a criterion for Δt to ensure the accuracy of the simulation. The dimensionless ratio

$$\frac{\langle R\rangle}{d} = \frac{1}{d}\sqrt{\frac{2k_{\text{B}}T}{\xi}\Delta t} \tag{21}$$

is used to select a proper Δt, where d is the spring length, and $\langle R\rangle$ is the average amplitude of the scalar random force (in the dimension of length) which is supposed to be much smaller than d. When $\langle R\rangle/d = 5\%$ and 2%,Δt can be determined from Eq. (21) as $0.00125\frac{\xi d^2}{k_{\text{B}}T}$ and $0.0002\frac{\xi d^2}{k_{\text{B}}T}$, respectively. These two time steps will be used in the example simulations discussed below.

3. Case study

To verify the effective bead–spring model, we have implemented the model in several example problems with theoretical predictions and experimental results.

3.1. Polymers with strong longitudinal stiffness subject to stretching forces at ends

Since no polymer has infinite longitudinal stiffness, it is more reasonable to model their behaviors with bead–spring systems. In this example, a 9-beads bead–spring chain with spring constant $k = 10000k_{\text{B}}T/d^2$ under extension is studied. The stretching forces $f^{\text{stretching}}$ at ends are ranged from $1k_{\text{B}}T/d$ to $2000k_{\text{B}}T/d$. To obtain accurate simulation results for verification, we first conduct traditional bead–spring model simulations with very small time step $\Delta t = 5\times10^{-6}\frac{\xi d^2}{k_{\text{B}}T}$, which makes $2\Delta tk/\xi = 0.1 \ll 1$ and Eq. (9) approximately satisfied. Fig. 2 shows the normalized mean end-to-end distance as the function of the stretching force, and the accurate results from traditional bead–spring model scheme are denoted with open circles. Two theoretical predictions based on freely jointed chain (FJC) model [19,20] and simple static stretching of bonds in the polymer backbone are also presented in the figure. When the longitudinal stiffness of polymers is strong and the stretching force weak, corresponding to the entropy-dominated situation, the FJC model, which is essentially a bead–rod model with the rod length d taken as the Kuhn length, can describe the chain behaviors quite well. When the FJC is subject to a pair of stretching forces $f^{\text{stretching}}$ at the ends, the theoretical force-extension relation can be written as follows [20]

$$|\langle \boldsymbol{x}_1 - \boldsymbol{x}_N\rangle|/L = \coth\frac{df^{\text{stretching}}}{k_{\text{B}}T} - \frac{k_{\text{B}}T}{df^{\text{stretching}}} \tag{22}$$

where $\boldsymbol{x}_1$ and $\boldsymbol{x}_N$ are the position vectors of the end beads and L is the chain length. If the stretching force $f^{\text{stretching}}$ becomes stronger and stronger, the bead–spring system will enter the bond-dominated regime, and the extension of the chain can be computed from simple static stretching. Obviously, this bond extension regime cannot be captured by bead–rod models in which the chain or rods are inextensible. Fig. 2 shows that the proposed second-order effective bead–spring model, denoted with solid squares, can accurately capture the

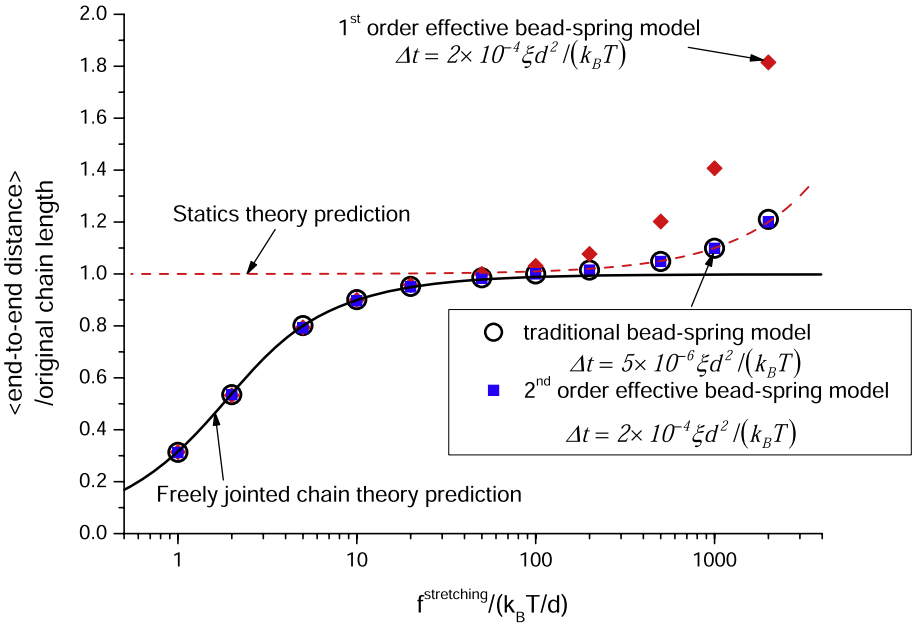

Fig. 2. The normalized end-to-end distance of a bead–spring chain with the spring constant $k = 10000k_BT/d^2$ as a function of the stretching force.

behaviors of this stiff bead–spring system in transition from entropy-dominated to bond-dominated regimes, also with much larger time step $\Delta t = 2 \times 10^{-4} \frac{\xi d^2}{k_B T}$, 40 times as that used in traditional bead–spring model in this case.

Therefore, the proposed effective bead–spring models are more efficient than the traditional bead–spring models and bead–rod models in that large time steps can be used and yet no matrix computation is required. Moreover, the second-order effective bead–spring model can surpass bead–rod models by simulating more general cases.

In addition, it is interesting to note that the first-order effective bead–spring model yields sufficiently accurate prediction for the stretched chain with the mean normalized end-to-end distance up to 90%, as denoted with solid diamonds in Fig. 2. We note that the first-order effective bead–spring model is essentially equivalent to the traditional bead–spring model except that the spring constant k is replaced with the effective spring constant $\tilde{k}$. Therefore, it should be easy to implement this scheme in the existing codes on bead–spring models and no code revision is needed. The error associated with the first-order effective bead–spring model can be approximately evaluated before the simulation. According to Eq. (11), the relative error δ on spring length due to the ignorance of $\tilde{f}$ can be estimated as

$$\delta = \frac{2\Delta t}{\xi d}\tilde{f} \tag{23}$$

Substituting Eq. (17) into the equation above gives

$$\delta = \frac{2\Delta t}{\xi d}\left[1 - \frac{\xi}{2\Delta t k}\left(1 - e^{-\frac{2\Delta t k}{\xi}}\right)\right]\langle f\rangle \tag{24}$$

The amplitude of $\langle f\rangle$ can often be estimated a priori. For instance, the mean value $\langle f\rangle$ in the present example should be comparable to the stretching force $f^{\text{stretching}}$, namely $\langle f\rangle \sim f^{\text{stretching}}$. Together with the parameters $\Delta t = 2 \times 10^{-4} \frac{\xi d^2}{k_B T}$ and $k = 10000k_BT/d^2$ used in the example, the estimate of the relative error is $\delta \sim 0.0003 f^{\text{stretching}}/(k_BT/d)$. Therefore, if $f^{\text{stretching}} = 10k_BT/d$ (under this pair of forces the bead–spring chain would be almost straight as shown in Fig. 2), the relative error of the spring length is only 0.3%.

We have also investigated the rigid-rod constraint in the FJC model by considering a bead–rod system consisting of nine beads subject to $f^{\text{stretching}} = k_B T/d$, which corresponds to the limiting case of $k \to \infty$. According to Eq. (14a), the effective spring constant $\tilde{k} = \frac{\xi}{2\Delta t}$ is used in the first-order effective bead–spring model. When $\Delta t = 0.00125\frac{\xi d^2}{k_B T}$, corresponding to $\langle R\rangle/d = 5\%$, the average spring/rod length equals $1.006d$ with a standard deviation of $0.075d$. When $\Delta t = 0.0002\frac{\xi d^2}{k_B T}$, corresponding to $\langle R\rangle/d = 2\%$, the average spring/rod length equals $1.0009d$, with a standard deviation of $0.03d$. After introducing the effective constraint force $\tilde{f} = \langle f\rangle$ into the second-order effective bead–spring model based on Eq. (17), the average spring/rod length becomes $1.0002d$ and $1.00002d$ for the two different time steps above, but the standard deviation remains the same. Therefore, the constraint conditions for the rod length can be approximately satisfied and smaller time step or smaller $\langle R\rangle/d$ would lead to stronger constraint. The effective bead–spring model seems to compromise between accuracy in rod length constraint and efficiency in the behaviors of polymers with moderate computational effort.

In addition, we have studied the dependence of the error on the number of beads. A 101-beads bead–rod chain under stretching forces is simulated by the effective bead–spring model, and no error accumulation effect with the chain length is found.

3.2. Polymers with small longitudinal stiffness

Another advantage of the effective bead–spring model is that relatively larger time steps can be taken in simulations to ensure accuracy or convergence for polymers with small longitudinal stiffness, corresponding to the traditional bead–spring model. To demonstrate this point, we consider a bead–spring chain with nine beads under a stretching force $f = k_B T/d$. The physical spring constant is chosen to be $k = 900 k_B T/d^2$. The normalized mean end-to-end distance, $|\langle \boldsymbol{x}_1 - \boldsymbol{x}_n\rangle|/L$ as defined in the previous example, should be independent of the simulation time step Δt. We investigate the sensitivity of the simulation results to time step in the traditional bead–spring model and in the proposed effective bead–spring model. Four time steps, $\Delta t \Big/ \left(\frac{\xi d^2}{k_B T}\right) = 1\times 10^{-4}, 1\times 10^{-3}, 1\times 10^{-2}$ and 1×10^{-1}, are considered. It has been verified that the smallest time step can ensure the accuracy of simulations. In the traditional bead–spring model, the simulations diverge for two larger time steps. The relative extensions $|\langle \boldsymbol{x}_1 - \boldsymbol{x}_n\rangle|/L$ are 0.316 and 0.286 for $\Delta t/\left(\frac{\xi d^2}{k_B T}\right) = 1\times 10^{-4}$ and 1×10^{-3}, respectively. In contrast, the simulations in the first-order effective bead–spring model converge for four time steps, and the relative extension are 0.316, 0.316, 0.33 and 0.52 for $\Delta t \Big/ \left(\frac{\xi d^2}{k_B T}\right) = 1\times 10^{-4}, 1\times 10^{-3}, 1\times 10^{-2}$ and 1×10^{-1}, respectively. The simulation results from the effective bead–spring model, as shown in Fig. 3, is less sensitive to time step and remains reasonably accurate even for $\Delta t/\left(\frac{\xi d^2}{k_B T}\right) = 1\times 10^{-2}$ at which the traditional bead–spring model already diverges.

While the results in Fig. 2 indicated that longitudinal elasticity dominates the deformation of polymer chains under very large stretching forces, those in Fig. 3 showed that, even for small tensile forces on the order of a few $k_B T/d$, longitudinal elasticity can play a significant role in the deformation of a polymer chain when its longitudinal stiffness is smaller than, say, 1000 $k_B T/d^2$.

3.3. Polymers with long-range hydrodynamic interactions

The hydrodynamic interactions are often important in Brownian dynamics simulations of polymer chains [4–6,14,17]. To demonstrate the effective bead–spring model for hydrodynamic interactions, we consider the translational diffusion coefficient of a DNA chain. The translational diffusion matrix $\mathbf{D}$ in Eq. (1) is given by Rotne–Prager hydrodynamic interaction tensor [21]. The DNA chain is modeled as a bead–rod system with the first-order effective bead–spring integration scheme. The computational parameters are taken as follows:

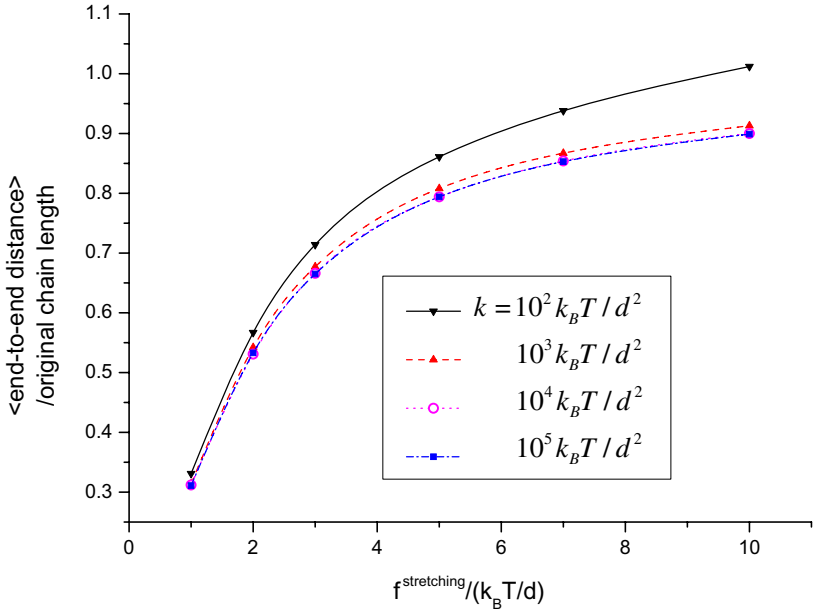

Fig. 3. The normalized end-to-end distance of a bead–spring chain with different spring constants $k = 10^2$, 10^3, 10^4, 10^5 (k_BT/d^2) as a function of the stretching force.

Persistence length $p = 53$ nm;
Rod length $d = 106$ nm;
Temperature $T = 293$ K;
Radius of beads $a = 0.07d$;
Viscosity of water $\eta = 0.001$ kg s^{-1} m^{-1};
Time step $\Delta t = 0.0002\frac{\xi d^2}{k_B T}$, where $\xi = 6\pi\eta a$.

The translational diffusion coefficient D_t can be estimated via a long trajectory simulation from the mean-square fluctuations of the system according to the Einstein–Stokes relation

$$6tD_t(t) = \langle \| \boldsymbol{x}_{\text{c.m.}}(t) - \boldsymbol{x}_{\text{c.m.}}(0) \|^2 \rangle \tag{25}$$

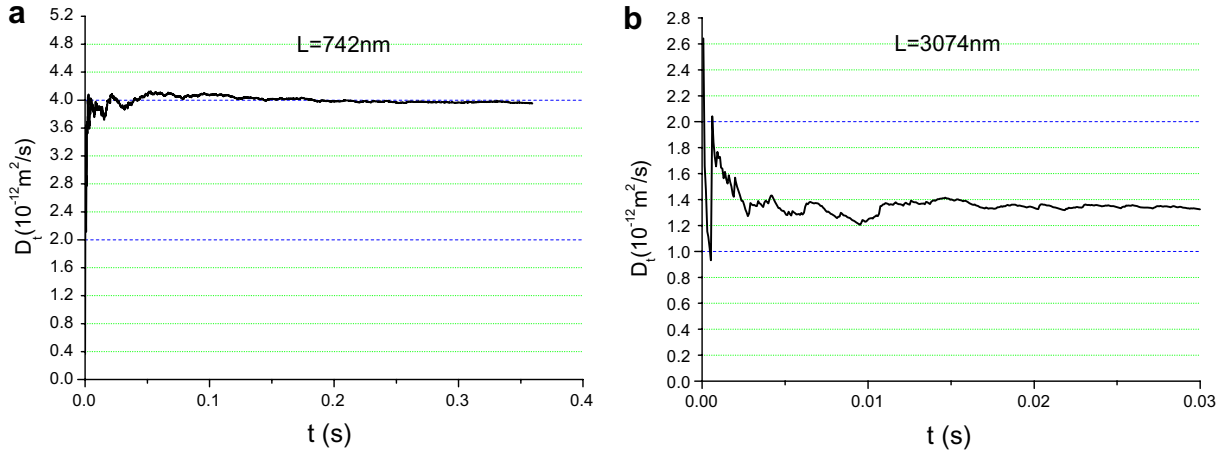

Fig. 4. Convergence of the translational diffusion coefficient D_t for a DNA chain with contour length (a) $L = 742$ nm and (b) $L = 3074$ nm.

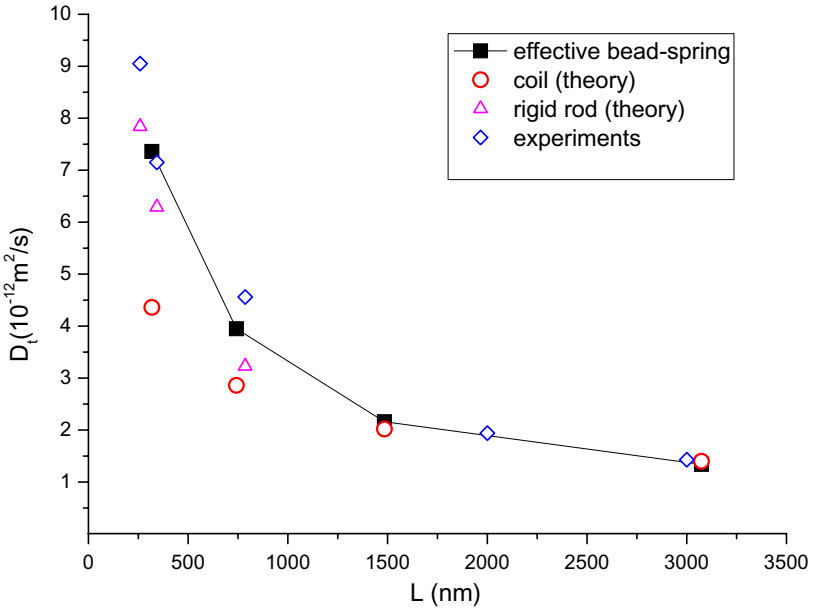

Fig. 5. Comparison of simulated translational diffusion coefficient D_t with corresponding theoretical predictions and experimental measurements.

where $\boldsymbol{x}_{\text{c.m.}}(t)$ is the center of mass of the chain at time t. Figs. 4(a) and (b) show that the converged translational diffusion coefficient D_t for DNA chain with different lengths can be obtained within a finite time span. The relation between D_t and the chain length from the simulation of the effective bead–spring model is shown in Fig. 5. For comparison, theoretical predictions and experimental results [22–24] are also drawn in Fig. 5, and good agreement among simulations, theory and experiments is observed.

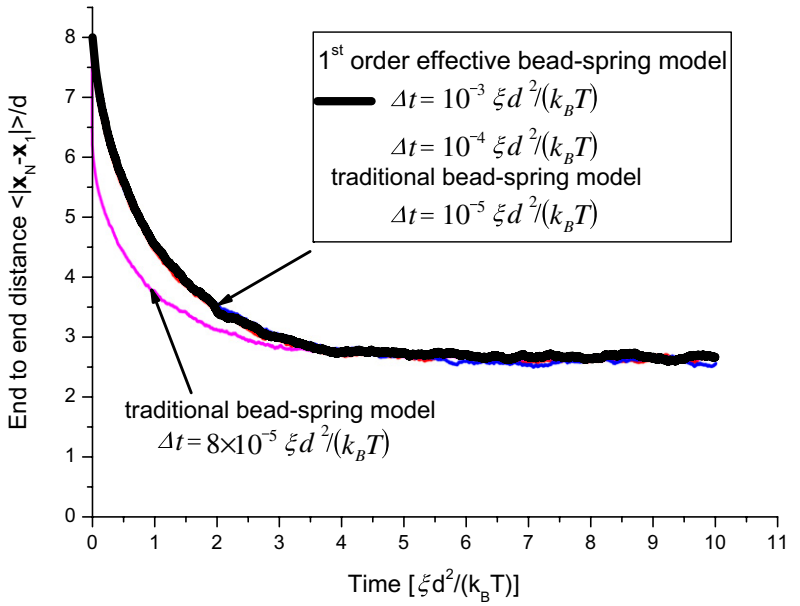

Fig. 6. Dynamic relaxation of an initially straight 9-bead chain. The normalized end-to-end distance of the chain as a function of the normalized time.

3.4. Dynamic relaxation of an initially straight chain

To demonstrate the proposed effective bead–spring model for real dynamic simulations, we study the conformational relaxation of a polymer chain from an initial straight configuration. To describe the relaxation dynamics, Brownian dynamics simulations are performed based on the proposed effective bead–spring model as well as the traditional bead–spring model. For a 9-bead chain with spring constant $k = 10000k_BT/d^2$, it can be seen from Fig. 6 that the computed trajectories based on the effective bead–spring model under time steps up to $\Delta t = 10^{-3}\xi d^2/(k_BT)$ show good agreement with those obtained from the traditional bead–spring model with a much smaller time step of $\Delta t = 10^{-5}\xi d^2/(k_BT)$. In contrast, even for a time step as small as $8 \times 10^{-5}\xi d^2/(k_BT)$, the trajectory based on the traditional bead–spring model differs significantly from the result at the time step of $10^{-5}\xi d^2/(k_BT)$. Further simulations indicate that the results based on the traditional bead–spring model diverge as soon as the time step exceeds $10^{-4}\xi d^2/(k_BT)$.

4. Comparisons with other integration algorithms

Besides the implicit integration scheme with rigid constraints shown in Eqs. (3) and (4), there are also a number of other implicit algorithms as discussed in Fixman's paper [25]. We have used the simple example in Fig. 1(a) to test and compare among all schemes. The governing equation for this single degree of freedom system in the absence of an external force is

$$\dot{x} = -\frac{2k}{\xi}x \tag{26}$$

with solution

$$x(t) = x(0)\mathrm{e}^{-\frac{2k}{\xi}t} \tag{27}$$

On the other hand, by integrating Eq. (26), the discrete numerical solution can be obtained as

$$x^{(n+1)} = x^{(n)} + \int_{t_n}^{t_{n+1}} \dot{x}dt = x^{(n)} - \frac{2k}{\xi}\int_{t_n}^{t_{n+1}} x\,\mathrm{d}t \tag{28}$$

by which various integration schemes can be tested and compared.

For the trapezoidal rule (or the midpoint rule) $\int_{t_n}^{t_{n+1}} x\,\mathrm{d}t \cong \Delta t(x^{(n+1)} + x^{(n)})/2$ in Fixman's paper [25], the resulting integration scheme is

$$x^{(n+1)} = \frac{1 - \frac{k\Delta t}{\xi}}{1 + \frac{k\Delta t}{\xi}}x^{(n)} \tag{29}$$

For the backward Euler rule $\int_{t_n}^{t_{n+1}} x\,\mathrm{d}t \cong \Delta t x^{(n+1)}$ in Fixman's paper [25], the integration scheme is

$$x^{(n+1)} = \frac{1}{1 + \frac{2k\Delta t}{\xi}}x^{(n)} \tag{30}$$

The traditional bead–spring scheme used in the previous section is essentially the Euler algorithm with $\int_{t_n}^{t_{n+1}} x\,\mathrm{d}t \cong \Delta t x^{(n)}$ and

$$x^{(n+1)} = \left(1 - \frac{2k\Delta t}{\xi}\right)x^{(n)} \tag{31}$$

There is also a second-order scheme obtained by iteration of the Euler approximation in the trapezoidal rule [25,26,4]. In this case, an auxiliary position $\tilde{x}^{(n+1)}$ is first computed according to the Euler algorithm in Eq. (31) as

$$\tilde{x}^{(n+1)} = \left(1 - \frac{2k\Delta t}{\xi}\right)x^{(n)} \tag{32}$$

Then the trapezoidal rule is used to estimate the integral

$$\int_{t_n}^{t_{n+1}} x\,\mathrm{d}t \cong \Delta t(\tilde{x}^{(n+1)} + x^{(n)})/2 \tag{33}$$

Solving Eqs. (28), (32) and (33) yields the second-order integration scheme

$$x^{(n+1)} = \left[1 - \frac{2k\Delta t}{\xi}\left(1 - \frac{k\Delta t}{\xi}\right)\right]x^{(n)} \tag{34}$$

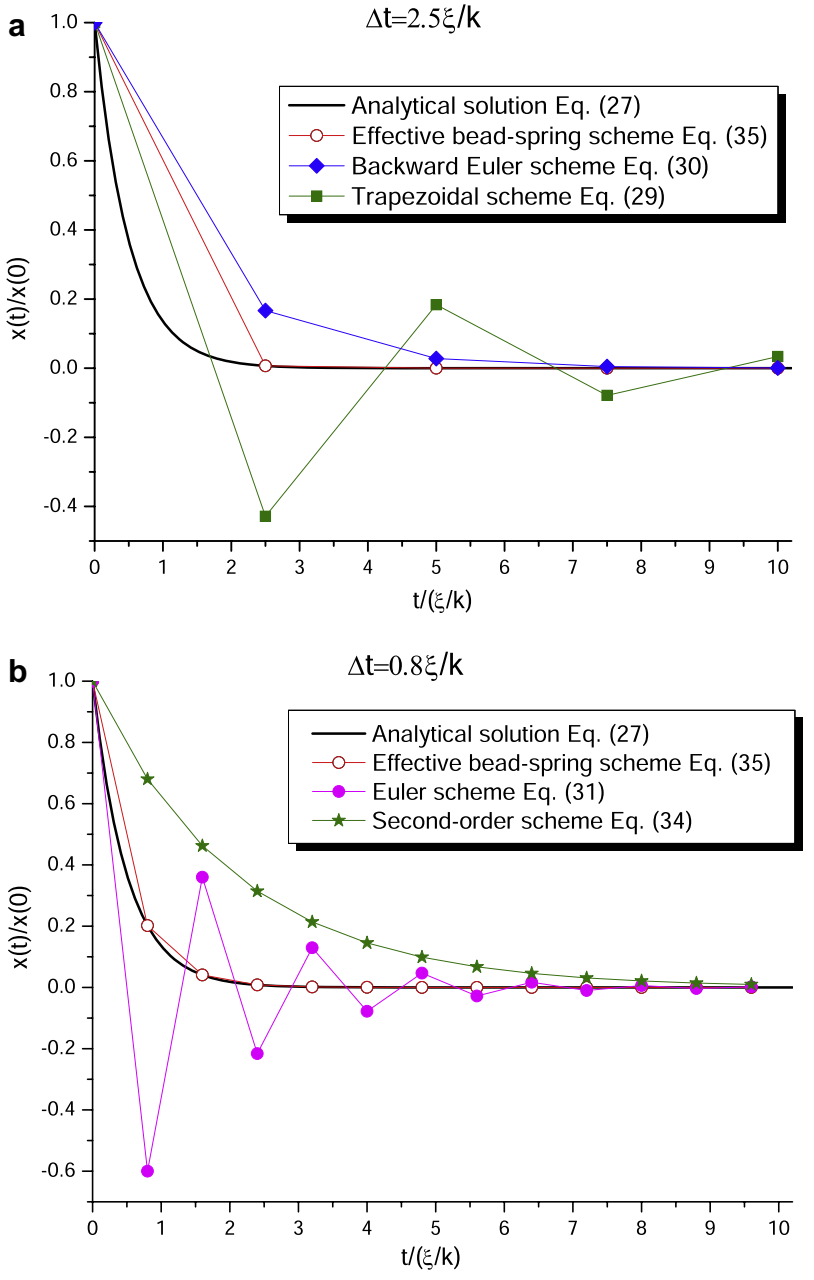

Fig. 7. The normalized displacement as a function of the normalized time for the single degree of freedom system in Fig. 1(a). Two time steps are used in various integration schemes: (a) $\Delta t = 2.5\xi/k$; (b) $\Delta t = 0.8\xi/k$.

It can be seen that all the schemes listed above have errors in computing $\int_{t_n}^{t_{n+1}} x\,\mathrm{d}t$ in Eq. (28), especially when the time step Δt is relatively large. However, the effective bead–spring model essentially uses the analytical solution Eq. (27) to compute this integral, i.e. $\int_{t_n}^{t_{n+1}} x\,\mathrm{d}t = \frac{\xi}{2k}x^{(n)}\left(1-\mathrm{e}^{-\frac{2k\Delta t}{\xi}}\right)$, leading to

$$x^{(n+1)} = \mathrm{e}^{-\frac{2k\Delta t}{\xi}}x^{(n)} = \left(1-\frac{2\tilde{k}\Delta t}{\xi}\right)x^{(n)} \tag{35}$$

which is the same as Eq. (11). In this simple example, the effective bead–spring scheme is consistent with the analytical solution of Eq. (27) for any time step.

Figs. 7(a) and (b) show the displacement as a function of time for this system. It is seen that, while the effective bead–spring algorithm follows the analytical solution exactly, all other schemes deviate from the analytical solution more or less, especially when the time step is large. The stability condition of an integration scheme can be obtained by investigating the ratio $|x^{(n+1)}/x^{(n)}|$. Only $|x^{(n+1)}/x^{(n)}| < 1$ leads to stable simulations. It can be shown that the Euler scheme and the second-order scheme are stable only when $\Delta t < \xi/k$, and are not shown in Fig. 7(a). In contrast, the effective bead–spring scheme, the trapezoidal scheme and the backward Euler scheme are unconditionally stable.

5. Summary and discussions

By generalizing an analytical solution in the case of a single pair of beads connected by spring, we have developed an efficient effective bead–spring model for simulations of polymer chains with arbitrary longitudinal stiffness. The model adopts a simple integration scheme as in traditional bead–spring models while taking large time steps as in bead–rod models. By simply replacing the spring constant in traditional bead–spring model with an effective spring constant, the first-order algorithm can be implemented with sufficient accuracy in the simulation of entropy-dominated polymer behaviors. To expand the applicability of the concept, by introducing additional effective constraint forces, we have further formulated a second-order effective bead–spring model which appears to be accurate in both entropy-dominated and bond-dominated regimes. Because of the efficiency and flexibility, the proposed effective bead–spring model may help simulate the dynamic behaviors of many types of polymer chains with different chain elasticity via an efficient unified integration scheme with large time steps. Combining with angular springs, this model can also be used to simulate the bending behaviors of semiflexible polymers.

Acknowledgement

BL acknowledges the support from Alexander von Humboldt Foundation in Germany and the National Natural Science Foundation of China through the Grants #10542001, #10702034 and #10732050.

References

[1] J. Kierfeld, O. Niampoly, V. Sa-yakanit, R. Lipowsky, Stretching of semiflexible polymers with elastic bonds, Eur. Phys. J. E 14 (2004) 17–34.
[2] J.F. Marko, E.D. Siggia, Stretching DNA, Macromolecules 28 (1995) 8759–8770.
[3] T. Odijk, Stiff chains and filaments under tension, Macromolecules 28 (1995) 7016–7018.
[4] H.M. Jian, A.V. Vologodskii, T. Schlick, Combined wormlike-chain and bead model for dynamic simulations of long linear DNA, J. Comput. Phys. 136 (1997) 168–179.
[5] S.A. Allison, Brownian dynamics simulation of wormlike chains-fluorescence depolarization and depolarized light-scattering, Macromolecules 19 (1986) 118–124.
[6] M. Doi, S.F. Edwards, The Theory of Polymer Dynamics, Oxford university press, New York, 1986.
[7] A. Baumgaertner, Simulation of polymer motion, Ann. Rev. Phys. Chem. 35 (1984) 419–435.
[8] R.B. Bird, R. Curtiss, R.C. Armstrong, O. Hassager, The Dynamics of Polymer Liquids, vol. II, Wiley, New York, 1987.
[9] E.S.G. Shaqfeh, G.H. McKinley, N. Woo, D.A. Nguyen, T. Sridhar, On the polymer entropic force singularity and its relation to extensional stress relaxation and filament recoil, J. Rheol. 48 (2004) 209–221.
[10] M. Somasi, B. Khomami, N.J. Woo, J.S. Hur, E.S.G. Shaqfeh, Brownian dynamics simulations of bead–rod and bead–spring chains: numerical algorithms and coarse-graining issues, J. Non-Newtonian Fluid Mech. 108 (2002) 227–255.

[11] C.C. Hsieh, S. Jain, R.G. Larson, Brownian dynamics simulations with stiff finitely extensible nonlinear elastic-Fraenkel springs as approximations to rods in bead–rod models, J. Chem. Phys. 124 (2006) 044911.
[12] M. Fixman, Simulation of polymer dynamics, I. General theory, J. Chem. Phys. 69 (1978) 1527–1537.
[13] T.W. Liu, Flexible polymer chain dynamics and rheological properties in steady flows, J. Chem. Phys. 90 (1989) 5826–5842.
[14] J. Wang, H. Gao, A generalized bead–rod model for Brownian dynamics simulations of wormlike chains under strong confinement, J. Chem. Phys. 123 (2005) 084906.
[15] J. Wang, X. Fan, H. Gao, Stretching short DNAs in electrolytes, Molecular & Cellular Biomechanics 3 (2006) 13–19.
[16] W. Kuhn, F. Grun, Relationships between elastic constants and stretching double refraction of highly elastic substances, Kolloid-Z 101 (1942) 248–271.
[17] D.L. Ermak, J.A. McCammon, Brownian dynamics with hydrodynamic interactions, J. Chem. Phys. 69 (1978) 1352–1360.
[18] B. Hess, H. Bekker, H.J.C. Berendsen, J.G.E.M. Fraaije, LINCS: a linear constraint solver for molecular simulations, J. Comput. Chem. 18 (1997) 1463–1472.
[19] P.J. Flory, Statistical Mechanics of Chain Molecules, Interscience Publishers, New York, 1969.
[20] F. Bueche, Physical Properties of Polymers, Interscience, New York, 1962.
[21] J. Rotne, S. Prager, Variational treatment of hydrodynamic interaction in polymers, J. Chem. Phys. 50 (1969) 4831–4837.
[22] H.T. Goinga, R. Pecora, Dynamics of low molecular weight DNA fragments in dilute and semidilute solutions, Macromolecules 24 (1991) 6128–6138.
[23] R. Pecora, DNA: a model compound for solution studies of macromolecules, Science 251 (1991) 893–898.
[24] D.E. Smith, T.T. Perkins, S. Chu, Dynamical scaling of DNA diffusion coefficients, Macromolecules 29 (1996) 1372–1373.
[25] M. Fixman, Implicit algorithm for Brownian dynamics of polymers, Macromolecules 19 (1986) 1195–1204.
[26] A. Iniesta, J.G. De La Torre, A 2nd-order algorithm for the simulation of the Brownian dynamics of macromolecular models, J. Chem. Phys. 92 (1990) 2015–2018.