J. Fluid Mech. (1995), *vol.* 282, *pp.* 373–403

Computer simulations of Brownian motion of complex systems

By P. S. GRASSIA[1], E. J. HINCH[1] AND L. C. NITSCHE[2]

[1]Department of Applied Mathematics and Theoretical Physics, The University of Cambridge, Silver Street, Cambridge CB3 9EW, UK

[2]Department of Chemical Engineering, The University of Illinois at Chicago, 810 South Clinton Street, Chicago, IL 60607, USA

(Received 3 December 1993 and in revised form 4 August 1994)

Care is needed with algorithms for computer simulations of the Brownian motion of complex systems, such as colloidal and macromolecular systems which have internal degrees of freedom describing changes in configuration. Problems can arise when the diffusivity or the inertia changes with the configuration of the system. There are some problems in replacing very stiff bonds by rigid constraints. These problems and their resolution are illustrated by some artificial models; firstly in one dimension, then in the neighbourhood of an ellipse in two dimensions and finally for the trimer polymer molecule.

1. Introduction

Computer simulations of Brownian motion are currently being used to study a wide range of phenomena in colloidal and macromolecular systems. With the growth in computing power, simulations were pioneered by Fixman (1978*b*), Ermak & McCammon (1978), Pear & Weiner (1979) and Helfand, Wasserman & Weber (1980) to study the relaxation and rheology of polymer chains. Now, in addition, simulations are being used to study the hydrodynamic thickness of a layer of polymer absorbed on a wall (Parnas & Cohen 1991), dynamic light scattering from a polymer chain (Rey, Freire & Garcia de la Torre 1989), the rheology of colloidal dispersions (Heyes & Melrose 1993; Rigos & Wilemski 1992; Sherwood 1992) and site-specific biochemical reactions (Luty *et al.* 1993; Nambi, Wierzbicki & Allison 1992; Northrup & Erickson 1992).

Diffusive Brownian motion describes the slow changes in configuration of complex colloidal and macromolecular systems. Systems of a few molecules change more rapidly and are more appropriately described by molecular dynamics simulations, which involve integrating Newton's equations of motion for each of the interacting molecules. For the slower diffusive changes of the configuration of larger systems, one can set the inertia to zero, which suppresses the rapid process of inertia-friction relaxation of momentum. In order to determine the correct description of the Brownian diffusion of some complex systems, we shall however find it necessary to consider the system before the mass is set to zero, because the limit of vanishing mass is a singular limit. In this case we shall use a Langevin description of the system, in which Newton's equations have an added random force representing the action of the thermal motions. A Lagrangian formulation of Newton's equations will sometimes be employed which will enable us to cast the Langevin equation in terms of generalized coordinates.

For simple systems with only a few degrees of freedom, the diffusive motion is best studied numerically by solving the appropriate partial differential equation for the probability distribution function in the space of possible configurations, a partial differential equation similar to the advection-diffusion equation for temperature. For more complex systems with many degrees of freedom, numerical simulation of random walks of the diffusive Brownian motion through the different configurations can be more economical. The computational trade-off is between the rapidly increasing storage requirements of the partial differential equation over many dimensions against the slowly improving $N^{-1/2}$ statistical errors when taking averages over N simulated random walks. In some special circumstances, such as a steady system with conservative forces, the probability distribution of configurations is known analytically. However in general, for example for transient phenomena and for systems with non-conservative external forces as occur in shearing flows, one must seek the probability distribution by numerical methods.

The numerical algorithm to simulate the diffusive random walks is straightforward in very simple systems. To any systematic motion driven by a steady force, one must add a random displacement at each time-step. The amplitude of these independent random displacements is chosen proportional to the square root of the size of the time-step, which ensures that the random walks have a variance which grows linearly in time independent of the size of the time-step. Quite small time-steps can therefore be necessary when there is a small spatial structure to be resolved. We review the basic algorithm in §2, along with questions of numerical accuracy and stability, and the question of statistical accuracy in taking averages over time.

When the Brownian system becomes large and complex, three problems can arise which require the basic algorithm to be modified. The purpose of this paper is to collect and to illustrate these problems. The first problem, which was noted and resolved by Ermak & McCammon in 1978, occurs when the diffusivity changes with the configuration. Unless a correction is made to the basic algorithm, configurations of high diffusivity are erroneously depleted as a result of vigorous random walks away from such configurations, compared with more feeble random walks returning from the surrounding configurations with a smaller diffusivity. There are several ways to modify the basic algorithm, with perhaps the easiest being a half-step scheme proposed by Fixman (1978*a*). In §3 we illustrate this problem and its resolution using a simple one-dimensional example with an artificially varying diffusivity.

A second problem concerns systems in which the mass or inertia changes with the configuration. Here it is physically correct that the more vigorous thermal motion should deplete configurations of lower mass. But the basic diffusivity algorithm cannot behave in this way because no inertia is incorporated into that algorithm. Clearly the Langevin description of the system, which does include inertia, must be examined in order to design an appropriate modification of the basic algorithm. We discuss this problem in §4, again illustrating it using a one-dimensional example with an artificially varying inertia.

The problems of varying diffusivity and varying inertia have their origin in the singular nature of the limit of vanishing mass. Now the diffusion results for Langevin simulations, as opposed to results on the faster inertia-friction relaxation time-scale, are independent of the overall magnitude of the inertia, just depending on the fractional changes. This is found so long as the magnitude is sufficiently small, e.g. so that damped oscillations remain overdamped. Thus a safe and economical numerical strategy would be to use the Langevin equation not with the real inertia which can be extremely small but with an artificially larger, yet still small, value which would permit larger time-steps.

A third and perhaps the most important problem in making computer simulations of Brownian motion of complex systems comes from very stiff bonds. Polymer molecules are typically modelled by a chain of beads joined by bonds which are freely jointed at each bead, with the length of the stiff bonds varying little. When investigating the rheological properties of the polymer, one is interested in large-scale configurational changes as the polymer uncoils and not the changes in length of the individual bonds. Computer simulations must however resolve stably and accurately these small fast changes in bond length, and this can require extremely small time-steps. In order to concentrate on the more interesting slow changes in configuration, one might be tempted to replace a very stiff bond by a rigid constraint. Unfortunately the Brownian motion of the configuration of a system with rigid constraints is different from that with stiff bonds; to the extent that even the steady probability distribution of the configurations can be different in thermodynamic equilibrium with no externally applied forces, see e.g. Fixman (1978*a*). Two sources of the difference can be identified: (i) a change in the volume of the phase space (Hassager 1974; Rallison 1979) which is reflected in an inertia which varies with configuration, and (ii) variations in the width of a potential well (Helfand 1979) which lead to variations around the constraint surface for the stiff systems. Theoretical studies, e.g. see Hinch (1994), have shown that a system with rigid constraints can be made to behave in Brownian motion like an analogous system with very stiff bonds by the application of a pseudo-potential force. Our principal illustration of this problem of stiff bonds will be with the trumbbell or trimer polymer molecule in §7.

In order to illustrate some of the issues in representing a very stiff element by a modified rigid element, we explore in §5 a simple artificial example of a particle moving around an ellipse. A two-dimensional example is the simplest which has a non-trivial problem remaining after one degree of freedom has been rigidified or frozen out. In the stiff form, our example will have a Brownian particle strongly attracted by a potential force towards the curve of an ellipse, while in the rigid form the particle can move only along the curve of the ellipse. Generally, and certainly in the case of the trimer, freezing out one degree of freedom makes the diffusivity and the inertia depend on the configuration. To endow our simple ellipse example with this property, we need to make the mass of the particle and its friction coefficient anisotropic, i.e. to have different values for the components of motion along the two axes of the ellipse. This artificial example is set up in §5.

Our first analysis of the rigidified artificial example uses a generalized coordinate, an angle, to describe the one-dimensional motion around the ellipse. In §6 we revert to a two-dimensional Cartesian description of the motion and use constraint forces to keep the particle on the ellipse. In more complex systems, such a description can be simpler than using generalized coordinates, for which there may indeed be no obvious choice. The constrained Cartesian description, however, introduces new problems in certain types of computer simulations, where we shall find it is essential for the random thermal forces to be subjected to the constraints in addition to the motion itself of the particles. We shall further see how different formulations of the same constraints correspond to the limits of different stiff systems, with different variations of the width of the potential well around the constraint surface.

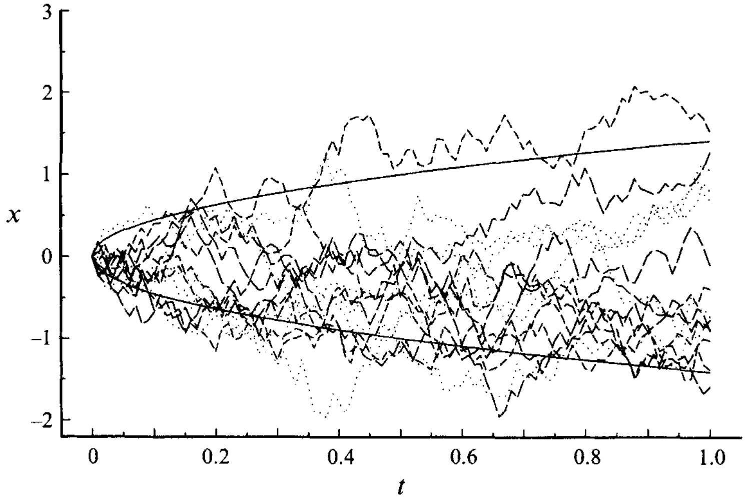

FIGURE 1. Random walks. Fifteen simulated diffusion random walks $x(t)$, starting at $t = 0$ from $x = 0$, with no systematic velocity $v^s = 0$ and with unit diffusivity $D = 1$. Five of the random walks have a time-step $\delta t = 10^{-2}$ (long dashes), five have $\delta t = 10^{-3}$ (short dashes) and five $\delta t = 10^{-4}$ (dots). The continuous curves are $\pm(2Dt)^{1/2}$.

2. The basic algorithm for Brownian motion

2.1. *Diffusion*

Consider an inertialess particle moving in one dimension at position x at time t. Assuming that friction is linear in the velocity, a steady force $f^s(x)$ will produce a steady or systematic velocity $v^s(x) = f^s/\zeta$, where ζ is the coefficient of friction. The diffusion coefficient of the Brownian motion is given by the Stokes–Einstein relation $D = kT/\zeta$, where kT is the Boltzmann temperature.

Computer simulations of Brownian walks can be made by adding at each time-step of δt a random displacement Ar_n to the systematic displacement $v^s\delta t$:

$$x_{n+1} = x_n + v^s\delta t + Ar_n .$$

Here r_n is a random number chosen with zero mean and independently of the random numbers at previous time-steps, and A is an amplitude selected so that the variance of the random part of the walk $\langle (x - x_0 - v^s(t - t_0))^2 \rangle$ grows linearly in time as $2D(t - t_0)$. Note that it is not necessary for the random numbers to have a Gaussian distribution. Note also that some standard random number generators do not have the required independence between adjacent numbers. As the commonly available random numbers are uniformly distributed on the interval [–0.5, 0.5], we assume that they have a variance $\langle r_n^2 \rangle = \frac{1}{12}$. It follows immediately that the amplitude of the random displacements is then given by

$$A = (24D\delta t)^{1/2} .$$

Figure 1 shows fifteen such simulated random walks $x(t)$, starting at $t = 0$ from $x = 0$, with no systematic velocity $v^s = 0$ and with unit diffusivity $D = 1$. Five of the random walks have a time-step $\delta t = 10^{-2}$, five have $\delta t = 10^{-3}$ and five $\delta t = 10^{-4}$. Viewed on an $O(1)$ time-scale, the random behaviour of the walks with the different time-steps is indistinguishable. Moreover the excursions grow in time like the solid curves $\pm(2Dt)^{1/2}$.

While the above forward time-stepping scheme is the simplest algorithm, higher-order schemes offer no improvements in accuracy because the changing random number creates a discontinuous velocity and the higher-order schemes require that the velocity and several of its derivatives be continuous before producing greater accuracy.

The generalization of the algorithm to higher dimensions is straightforward. For a particle at $x_i(t)$ moving with a tensor coefficient of linear friction ζ_{ij}, the steady forces f_i^s will produce a systematic velocity $v_i^s = (\zeta^{-1})_{ij} f_j^s$, while the Stokes–Einstein relation gives the diffusivity tensor $D_{ij} = kT(\zeta^{-1})_{ij}$. The basic algorithm for random Brownian walks then becomes

$$x_i^{n+1} = x_i^n + v_i^s \delta t + A_{ij} r_j^n \,.$$

Here r_j^n is a random vector with each component having zero mean and variance $\frac{1}{12}$ and being generated independently of the other components and independently of previous times. The tensor of the amplitudes A_{ij} must satisfy

$$A_{ik} A_{jk} = 24 D_{ij} \delta t \,.$$

Thus A is in some sense a square root of D. For the symmetric positive definite tensor D_{ij}, the required square root can always be obtained as a Cholesky decomposition, although an iterative method updating from the previous time-step can be faster numerically if the system is evolving slowly, see Fixman (1986). In the trimer problem of §7, an explicit expression can be written down.

2.2. *Langevin description*

The above diffusion description of the Brownian motion has the mass of the particle set to zero. This is a singular limit which in some complex systems produces erroneous results. In such circumstances when we need to examine the motion of the particle with a mass $m \neq 0$, we shall use a Langevin description, which is simply Newton's equation of motion for the particle including inertia, linear friction, steady forces and a random force $f^r(t)$ which drives the thermal motion of the particle

$$m\ddot{x} + \zeta \dot{x} = f^s + f^r \,.$$

The random force is assumed to be White Noise, i.e. has a correlation time shorter than any process of interest. The amplitude of the random force is then given by the fluctuation-dissipation theorem (see e.g. Kubo, Toda & Hashitsume 1985)

$$\langle f^r(t_1) f^r(t_2) \rangle = 2kT\zeta\delta(t_1 - t_2) \,.$$

To construct a computer simulation of the Langevin description, the delta function of the White Noise needs to be replaced by a numerical representation

$$\delta(t_1, t_2) = \begin{cases} 1/\delta t & \text{if } t_1 \text{ and } t_2 \text{ are in the same time-step of } \delta t \\ 0 & \text{otherwise} \end{cases}$$

For the simplest first-order forward time-stepping scheme, this leads to

$$\begin{aligned} x_{n+1} &= x_n + \dot{x}_n \delta t, \\ \dot{x}_{n+1} &= \dot{x}_n + m^{-1} \left(-\zeta \dot{x}_n + f^s + (24kT\zeta/\delta t)^{1/2} r_n \right) \delta t \end{aligned}$$

where r_n is again a random number of zero mean and variance $\frac{1}{12}$ chosen independently at each time-step. Figure 2 shows fifteen random walks $x(t)$ generated by

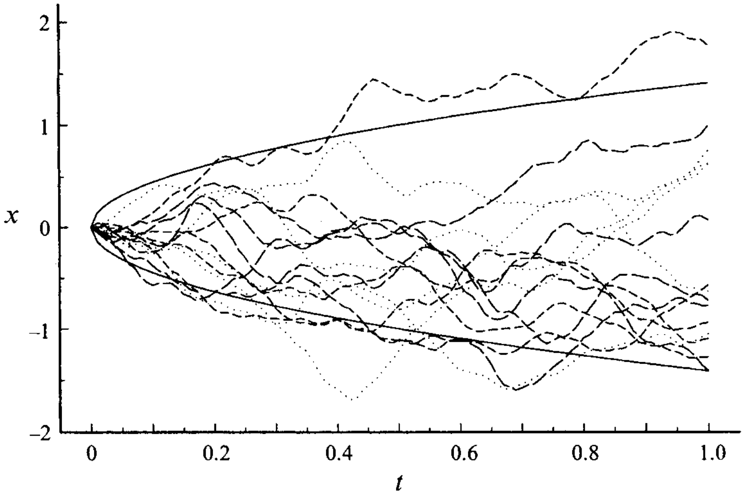

FIGURE 2. Random Langevin walks. Fifteen random walks $x(t)$ generated by the Langevin algorithm, each starting at $t = 0$ with $x = 0$ and $\dot{x} = 0$, with no steady force $f^s = 0$, with unit friction $\zeta = 1$ and unit thermal temperature $kT = 1$ which gives a unit diffusivity $D = 1$, and with a fairly small mass $m = 0.05$. Five of the random walks have a time-step $\delta t = 10^{-2}$ (long dashes), five have $\delta t = 10^{-3}$ (short dashes) and five $\delta t = 10^{-4}$(dots). The continuous curves are $\pm(2Dt)^{1/2}$.

this algorithm, each starting at $t = 0$ with $x = 0$ and $\dot{x} = 0$, with no steady force $f^s = 0$, with unit friction $\zeta = 1$ and unit thermal temperature $kT = 1$ which gives a unit diffusivity $D = 1$, and with a fairly small mass $m = 0.05$. Five of the random walks have a time-step $\delta t = 10^{-2}$, five have $\delta t = 10^{-3}$ and five $\delta t = 10^{-4}$. As in the earlier diffusive simulations without mass, viewed on an $O(1)$ time-scale the random behaviour of the walks with the different time-steps is indistinguishable, and the excursions grow in time like the solid curves $\pm(2Dt)^{1/2}$. The non-zero mass makes the individual curves smoother, with a persistence time for the velocity of $m/\zeta = 0.05$, which is most easily seen in figure 2 near $t = 0$ where $\dot{x} \approx 0$ until $t = 0.05$.

The extension to higher dimensions is similar to the extension of the diffusion algorithm and is straightforward, so we shall omit the details.

It is worth observing that, if one sets the mass equal to zero in the Langevin equation but keeps the random force as given by the numerical representation of the fluctuation-dissipation theorem, one obtains the diffusion algorithm in the form

$$x_{n+1} = x_n + \zeta^{-1}\left(f^s + (24kT\zeta/\delta t)^{1/2}\, r_n\right)\delta t\,.$$

This alternative form of the random diffusion displacements will be helpful in complex systems in which the friction and hence diffusivity change with configuration.

2.3. *Time averages*

One of the serious disadvantages of a computer method based on simulations of random walks is the slow and unavoidable $N^{-1/2}$ improvement in the statistical errors when taking averages over N independent random walks. In this paper we shall be calculating only steady-state equilibrium probability distributions. For such steady states it is often convenient to simulate a single random walk and form an average over a very long time. The question then arises of how long one must wait before the errors in the time-average are smaller than some desired tolerance ϵ. The answer is that one must wait until $t = t_0/\epsilon^2$, where t_0 is the time for the system to forget

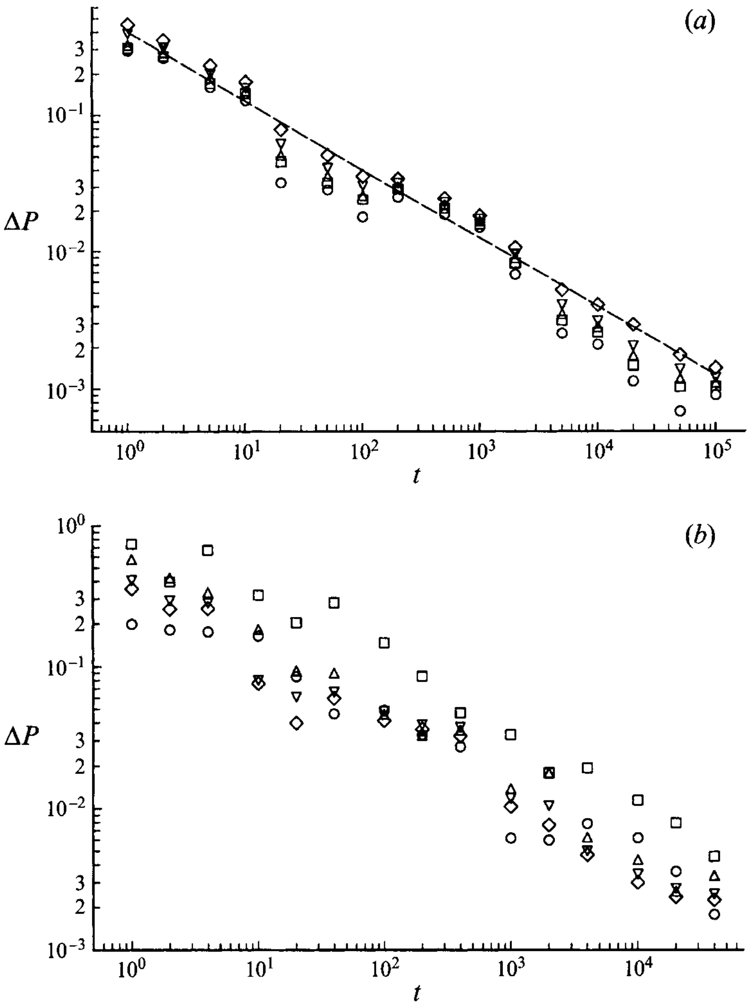

FIGURE 3. Statistical accuracy of time averages. The r.m.s. error ΔP in the computed probability distribution as a function of the time t over which time averages were made, for a particle diffusing in one dimension on the unit interval $[0,1]$ with periodic end conditions, with a unit diffusivity $D = 1$, with no systematic velocity $v^s = 0$, and using the basic diffusion algorithm with a time-step $\delta t = 10^{-3}$. The probability distribution is evaluated with different numbers of bins 5 (○), 10 (□), 20 (△), 50 (▽) and 100 (◇). The dashed line is $\Delta P = 0.4t^{-1/2}$. (*b*) The results for simulations which include mass, $m = 0$ (○), $m = 1$ (□), $m = 0.05$ (△), $m = 0.01$ (▽) and $m = 0.001$ (◇), all with time-step $\delta t = 2 \times 10^{-5}$.

the initial conditions. For diffusive Brownian motion, this time is that required to diffuse through all configurations. Thus for a particle diffusing in a one-dimensional box of length l with diffusivity D, $t_0 = O(l^2/D)$. When the mass is non-zero, there is also the inertia-friction relaxation time of the velocity m/ζ. Because however we are interested in colloidal and macromolecular systems in which the non-zero mass is small, this relaxation time tends to be smaller than the time to diffuse through all configurations, a comparison which defines the smallness of the mass.

Figure 3 shows how the statistical accuracy improves when averages are taken over progressively longer times. We simulated a particle diffusing in one dimension on the unit interval [0, 1] with periodic end conditions, with a unit diffusivity $D = 1$, with no systematic velocity $v^s = 0$, and using the basic diffusion algorithm with a time-step $\delta t = 10^{-3}$. From the position of the particle at the end of each time-step, we calculated in which of K ($K = 5, 10, 20, 50, 100$) equally spaced bins the particle was situated, and incremented the count of that bin. At the times plotted in figure 3, we calculated our numerical approximation to the probability distribution for the position of the particle by dividing the number of counts in each bin by the total number of counts, i.e. the total number of time-steps $t/\delta t$, and then dividing by the width of the bin $1/K$. The error in our numerical approximation to the probability distribution, which we have plotted in figure 3 as ΔP, is the root-mean-square of the difference between the approximation and the known exact uniform distribution. We see that the error decreases like $0.4t^{-1/2}$, corresponding for $l = 1$ and $D = 1$ to the memory time of the initial conditions $t_0 = 0.16l^2/D$. Thus one can obtain 1% accuracy by simulating to $t = 1500$, and 0.2% accuracy by continuing to $t = 4 \times 10^4$. It should be noted that the accuracy is independent of the number of bins, so long as they are sufficiently wide for the particle to have a reasonable chance of landing in each bin during t_0.

Figure 3(*b*) shows the improving accuracy with progressively longer time-averages for simulations which include mass. It is seen that for small masses $m < 1$, which is the case of practical interest, there seems to be an additional statistical error that scales roughly like $1.6m^{1/2}t^{-1/2}$. This means that one needs quite small masses, say $m < 0.07$, in order to keep this error less than twice the massless value. Thus for $m = 0.05$ we can obtain a 1% error by simulating to $t = 5000$, and a 0.2% error by continuing to $t = 4 \times 10^5$.

2.4. *Spatial resolution*

In addition to the statistical errors from taking averages over a small number of independent random walks, errors can occur due to inadequate spatial resolution. The random displacement in one time-step is $\delta x = O(2D\delta t)^{1/2}$. Requiring this to be smaller than some spatial structure of interest can require the time-step to be very small indeed.

We illustrate this problem in figure 4. Here we have made a simulation of a particle diffusing in one dimension on the unit interval [0, 1] with periodic end conditions, with unit diffusivity $D = 1$, and with a systematic velocity $v^s = \sin 2\pi x$ which produces a non-uniform equilibrium probability distribution

$$P(x) \;=\; 0.9937\,\mathrm{e}^{-\cos 2\pi x/2\pi}\,.$$

This distribution has a spatial structure of size say 0.5. Figure 4(*a*) gives the probability distributions obtained from numerical simulations of the random walks using various sizes of time-step ($\delta t = 0.03$, 0.01, 0.003 and 0.001). The simulations were taken to $t = 10^5$ so that statistical errors in the results are less than 0.2%. Quite clearly the time-step $\delta t = 0.03$, with the large random displacement $(2D\delta t)^{1/2} = 0.24$, smears out the details which are to be resolved. Figure 4(*b*) gives the root-mean-square errors in the computed probability distribution as a function of the size of the time-step. We see that the error is linear in δt, and that to obtain a 1% accuracy the time-step must be smaller than 0.003 in our particular example.

Figure 4(*b*) shows that the numerical error is first order in the time-step. This first-order error comes from the random part of the displacement $\delta x = O(2D\delta t)^{1/2}$ and should not be confused with the first-order error of the forward time-stepping scheme. Taking excessively large spatial steps does not produce any error in the probability flux if the probability distribution is uniform in space or if it varies linearly in space

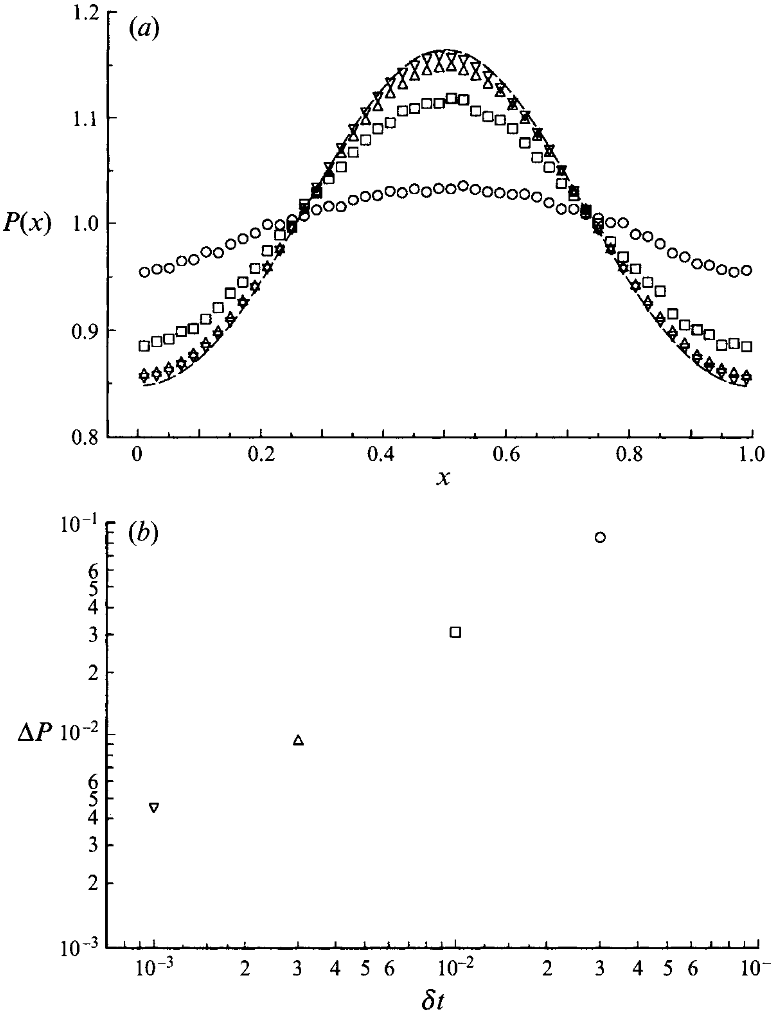

FIGURE 4. Time-step and spatial resolution. The spatial resolution as a function of the size of the time-step with $\delta t = 0.03$ (○), 0.01 (□), 0.003 (△) and 0.001 (▽), for a particle diffusing in one dimension on the unit interval [0, 1] with periodic end conditions, with a unit diffusivity $D = 1$, with the systematic velocity $v^s = \sin 2\pi x$, and using the basic diffusion algorithm and averaging over time 10^5. (*a*) The computed probability distributions $P(x)$, with the dashed curve being $0.9937 \exp(-\cos 2\pi x/2\pi)$. (*b*) The root-mean-square errors as a function of the size of the time-step δt.

(with a constant diffusivity): it is the quadratic part of the spatial variation of the probability distribution which is smeared out by large steps. Hence we expect errors $O((\mathrm{d}^2P/\mathrm{d}x^2)\,\delta x^2)$, and hence errors $O((\mathrm{d}^2P/\mathrm{d}x^2)\,2D\delta t)$, i.e. first order in the time-step. We note that this error, associated with the large random part of the displacement, would not be reduced by employing higher-order time-stepping schemes which would have the same large random displacements.

When simulations are made of systems which include a stiff spring, one can use as a monitor of the numerical accuracy the fact that the average potential energy stored in the spring should be $\frac{1}{2}kT$. Similarly for simulations with non-zero mass, one can check that the average of the kinetic energy is $\frac{1}{2}kT$ for each degree of freedom.

2.5. *Numerical stability*

The size of the time-step can also be restricted by possible numerical instabilities. One instability occurs with the basic diffusion algorithm of §2.1 when the steady force varies in space. We shall be interested in stiff springs with a steady force given by $f^s = -\kappa x$, with spring constant $\kappa \gg 1$. This produces a steady velocity $v^s = -\kappa x/\zeta$. For this velocity, the basic forward time-stepping algorithm is numerically stable only if $\kappa\delta t/\zeta < 1$. This criterion can be thought of as a restriction on the size of the time-step or a restriction on the stiffness of the spring. The criterion is in fact identical to the accuracy criterion in the previous section, because the present force law has an equilibrium probability distribution $P \propto \mathrm{e}^{-\kappa x^2/2kT}$, which leads to errors $O((\kappa/kT)2D\delta t) = O(2\kappa\delta t/\zeta)$. Thus implicit time-stepping methods, which would be more stable, would suffer from the same accuracy problems and so we will not use them.

In simulations of the Langevin equation by the method in §2.2, there is additionally an inertia-friction relaxation time of the velocity on the time-scale m/ζ. The forward time-stepping scheme therefore always has the restriction $\zeta\delta t/m < 1$. Again this criterion can be thought of as a restriction on the size of the time-step or a restriction on the smallness of the mass.

The numerical stability is a little more complicated for the Langevin equation when we consider stiff springs and small masses. The case of an overdamped oscillator is the case of practical interest in real colloidal and macromolecular systems. In the overdamped case, the inertia-friction relaxation of velocity on the time-scale m/ζ is much faster than the friction-spring relaxation of position on a time-scale ζ/κ, so that the numerical stability of the algorithm is given by $\zeta\delta t/m < 1$. In the underdamped case, there is a fast inertia-spring oscillation on a time-scale $(m/\kappa)^{1/2}$ and a slow frictional damping on a time-scale m/ζ, which produces a numerical stability criterion $(\kappa/m)^{1/2}\,\delta t < 1$. Note that when we consider stiff springs and small masses, the condition of overdamping requires $m/\zeta < \zeta/\kappa$, i.e. the mass must be smaller than the extent to which the stiffness is large.

3. Variable diffusivity

In complex Brownian systems, the diffusivity can change with the configuration of the system. This occurs when there are rigid constraints and also when there are hydrodynamic interactions.

3.1. *An erroneous algorithm*

The basic algorithm produces erroneous results when the diffusivity is not constant. We illustrate this in figure 5. Here we have made a simulation of a particle moving in one dimension on the unit interval [0, 1] with periodic end conditions, with no systematic velocity $v^s = 0$, and with a varying diffusivity given by

$$D(x) = \tfrac{1}{2} + \tfrac{1}{2}\cos^2 \pi x\,.$$

While there is no real physical system which has this particular spatial variation of the diffusivity, the simple example is sufficient to illustrate the problem and its resolution. Now the probability distribution of the position of the particle should be uniform. The simulation using the basic algorithm produces, however, a non-uniform distribution as shown by the ○ points in figure 5. All the simulations in this §3 used a small time-step $\delta t = 6.67 \times 10^{-5}$ and average data over a time 10^3, so that the statistical errors should be about 1%.

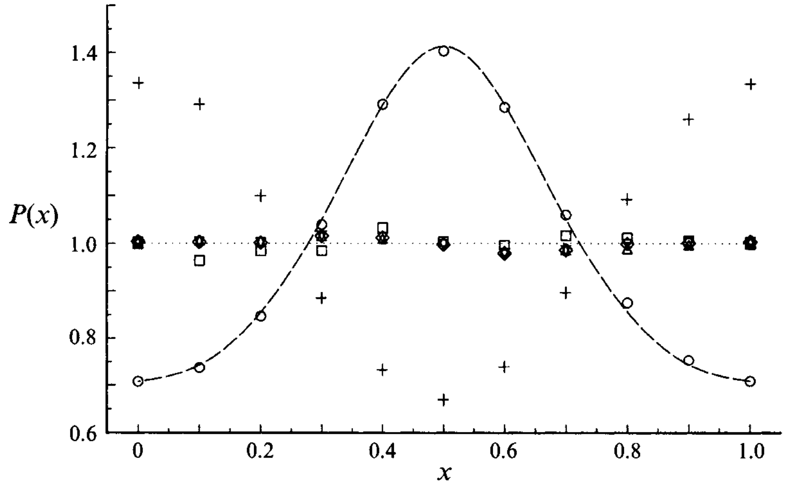

FIGURE 5. Variable diffusivity. The probability distributions from various algorithms for a particle diffusing on [0, 1] with variable diffusivity $D(x) = \frac{1}{2} + \frac{1}{2}\cos^2 \pi x$. All the simulations have a time-step of $\delta t = 6.67 \times 10^{-5}$ and average over a time 10^3. The simple diffusion algorithm of §2.1 (○). The dashed curve is the distribution $\propto 1/D(x)$. The Langevin algorithm with $m = 0.05$ (□). The Ermak & McCammon algorithm (△). The mid-point algorithm (▽). The mid-point algorithm plus the Ermak & McCammon $\nabla \cdot \boldsymbol{D}$ term (+). The two-step algorithm of §3.5 (◇).

The cause of the error is that the random walk has large random displacements where the diffusivity is large, near $x = 0$ and $x = 1$, and small random displacements where the diffusivity is small, near $x = \frac{1}{2}$. Thus according to the algorithm, regions of high diffusivity are depleted by the more vigorous walks away compared with the more feeble walks returning from the surrounding regions of lower diffusivity. In fact it can be shown that the basic algorithm should produce a probability distribution $\propto 1/D(x)$ in this one-dimensional example, and this distribution is given by the dashed curve in figure 5.

3.2. *Langevin description*

The resolution of the problem requires one to recognise that the limit of vanishing mass is a singular limit. Also plotted in figure 5 with the □ points is the result of a simulation of the Langevin equation with a small mass $m = 0.05$, with unit temperature $kT = 1$ and varying friction $\zeta(x) = 1/(\frac{1}{2} + \frac{1}{2}\cos^2 \pi x)$ which gives the same artificially varying diffusivity as above, and using the simple forward time-stepping algorithm of §2.2. This simulation with non-zero mass produces the correct uniform probability distribution. Note that the same result would have been found for any non-zero value of the mass so long as it is sufficiently small so that the inertial-friction relaxation time m/ζ is smaller than the diffusion time l^2/D.

3.3. *Ermak & McCammon algorithm*

An analysis of small nonlinearities due to the spatial variation of the friction in the Langevin equation shows (Fixman 1978*a*; Ermak & McCammon 1978) that the random walk has a mean drift with a mean velocity $\nabla \cdot \boldsymbol{D}$ in addition to any systematic velocity.

The analysis starts from the Langevin equation with varying friction

$$m_{ij}\ddot{x}_j + \zeta_{ij}(x)\dot{x}_j = f_i^s + f_i^r(t).$$

Now typical Brownian displacements $(kT/m)^{1/2} \times m/\zeta$ are always much smaller that the length scale over which the friction varies significantly, see e.g. Hinch & Nitsche (1993, § 6.4). One can therefore expand in the small amplitude of the Brownian motion

$$x(t) \;=\; x^{(0)} + x^{(1)}(t) + x^{(2)}(t) + \cdots .$$

Here $x^{(1)}(t)$ represents the standard linear Brownian motion, obtained by replacing $\zeta(x(t))$ by $\zeta(x^{(0)})$, and including the systematic motion from the steady forces f_i^s. The small nonlinear correction $x^{(2)}(t)$ results from the small variation of the friction between $\zeta(x^{(0)})$ and $\zeta(x(t))$. At second order one finds the governing equation

$$m_{ij}\ddot{x}_j^{(2)} + \zeta_{ij}(x^{(0)})\dot{x}_j^{(2)} + x_k^{(1)}\frac{\partial \zeta_{ij}}{\partial x_k}(x^{(0)})\dot{x}_j^{(1)} \;=\; 0\,.$$

This equation shows that after a few inertia-friction relaxation times, i.e. on a time scale much shorter than that required for the particle to diffuse a significant distance, a mean velocity is established. Denoting an average by an overbar, and using the Stokes-Einstein result of linear Brownian motion theory that the diffusivity of the random walk $\overline{x_i^{(1)}\dot{x}_j^{(1)}} \equiv D_{ij} = kT\zeta_{ij}^{-1}$, we find

$$\overline{\dot{x}_i^{(2)}} \;=\; -\zeta_{ij}^{-1}\frac{\partial \zeta_{jk}}{\partial x_l}\overline{x_l^{(1)}\dot{x}_k^{(1)}} \;=\; -\zeta_{ij}^{-1}\frac{\partial \zeta_{jk}}{\partial x_l}kT\zeta_{lk}^{-1} \;=\; \frac{\partial D_{ij}}{\partial x_j}\,.$$

Note that mean drift is established on a time-scale of the inertia-friction relaxation, and so cannot be derived from a consideration of a purely diffusive description with zero mass. The mean drift is required to convert the dispersion equation, which governs probability distribution functions, into the more familiar diffusion equation, see e.g. §7.2 Hinch & Nitsche (1993).

Incorporating the mean drift into the basic algorithm produces a modified diffusion algorithm

$$x_{n+1} \;=\; x_n + \left(v^s(x_n) + \frac{\mathrm{d}D}{\mathrm{d}x}(x_n)\right)\delta t + (24D(x_n)\delta t)^{1/2}\,r_n\,.$$

A simulation of our artificial one-dimensional problem using this algorithm does indeed produce the correct uniform distribution, as shown by the Δ points in figure 5.

Fixman (1978*a*) and Ermak & McCammon (1978) also provided the modified diffusion algorithm for higher dimensions

$$x_i^{n+1} = x_i^n + \left(v_i^s + \frac{\partial D_{ij}}{\partial x_j}\right)\delta t + A_{ij}r_j^n$$

with $A_{ik}A_{jk} \;=\; 24D_{ij}\delta t$ and with $\boldsymbol{v}^s$, $\nabla\cdot\boldsymbol{D}$ and $\boldsymbol{A}$ all evaluated at the beginning of the time-step with $\boldsymbol{x}^n$. (This algorithm is one of the Îto formulations of the random process, see e.g. Øksendal (1985).)

3.4. *Midpoint algorithm*

The disadvantage of the Ermak & McCammon (1978) algorithm is that one needs to know the spatial derivatives of the diffusivity and this can be a large task in a complex Brownian system. Fortunately there is an elegant alternative which was suggested by Fixman (1978*a*; which seems to have been overlooked despite being in the same issue of the journal which contained the paper by Ermak & McCammon). The idea is to replace the simple forward time-stepping scheme by the mid-point scheme. In fact any higher-order scheme would work, although the more sophisticated methods cannot produce improvements in accuracy because of the discontinuities in the random velocities. The mid-point method is applied to the diffusion formulation given at the end of §2.2. During the interval of the time-step, the random force is held constant at $(24kT\zeta(x_n)/\delta t)^{1/2}\, r_n$ while the premultiplying inverse friction is allowed to vary with the changing position $\zeta^{-1}(x(t))$. Thus the mid-point algorithm is

$$\begin{aligned} x_* &= x_n + \tfrac{1}{2}\zeta^{-1}(x_n)\left(f^s(x_n) + (24kT\zeta(x_n)/\delta t)^{1/2}\, r_n\right)\delta t, \\ x_{n+1} &= x_n + \zeta^{-1}(x_*)\left(f^s(x_*) + (24kT\zeta(x_n)/\delta t)^{1/2}\, r_n\right)\delta t\,. \end{aligned}$$

It is a simple exercise to check that this algorithm has the correct mean drift, $\mathrm{d}D/\mathrm{d}x$. A simulation of our artificial one-dimensional problem using this algorithm does produce the correct uniform distribution, as shown by the ∇ points in figure 5. The extension of this algorithm to higher dimensions is easy and straight forward, so we shall omit the details. (This algorithm is similar to a Stratonovich formulation of the random process, except that the random forces are held constant over the full time-step, and this leads to twice the drift compared with a strict Stratonovich formulation.)

We note that it would be wrong to add Ermak & McCammon's $\nabla\cdot\boldsymbol{D}\delta t$ term to the above mid-point algorithm, because the mean drift would then be twice the required value. That such an algorithm produces the wrong results is shown by the + points in figure 5 (which can be shown to be $P(x) \propto D(x)$ in this example).

3.5. *Another algorithm*

It is important in the mid-point algorithm to keep the random force constant while allowing the friction to vary. This may appear inconsistent, because the expression for the random force does include the friction. It is possible to construct an algorithm in which the frictional part of the random force does vary, namely

$$\begin{aligned} x_* &= x_n + v^s(x_n)\delta t + (24D(x_n)\delta t)^{1/2}\, r_n, \\ x_{n+1} &= x_n + v^s(x_*)\delta t + (24D(x_*)\delta t)^{1/2}\, r_n \end{aligned}$$

but this algorithm does have some undesirable features. First, taking the whole step twice is unusual and without the random part would have an $O(\delta t^2)$ local truncation error. Second, the algorithm cannot be generalized to higher dimensions because the generalization has the wrong tensorial structure. However in one dimension one can check that this alternative algorithm does have the correct mean drift, and this is borne out by a successful simulation as shown by the $\diamond$ points in figure 5.

4. Variable inertia

In complex Brownian systems, the inertia can change with the configuration of the system. This often occurs when there are rigid constraints. This can also occur when a Cartesian coordinate is replaced by a generalized coordinate, as we shall see later in a simple one-dimensional example. It also occurs when one includes the fluid virtual mass (Hinch & Nitsche 1993).

4.1. *The problem*

In regions where the mass is small, the thermal velocity will be compensatingly large in order to maintain the thermal kinetic energy at $\tfrac{1}{2}kT$ per degree of freedom. Regions of vigorous thermal velocity are depleted in favour of regions of higher mass with a feeble thermal activity. In the case of inertia, unlike the variable diffusivity, the phenomenon is correct: when a Maxwell–Boltzmann distribution in phase-space is projected onto the configuration-space by integrating out the momentum coordinates, there results a factor involving variations in the inertia, a metric factor measuring the size of the projected momentum-space (Hassager 1974; Rallison 1979).

The dependence of the Brownian motion on changes in inertia presents a problem to the diffusion algorithms of §3 which include no inertial terms and must therefore produce erroneous results. This is another example of the singular nature of setting the mass to zero in Brownian motion. To find out how to correct the diffusion algorithms, we need to examine the Langevin equation with varying inertia.

4.2. *Mean drift*

To write down the correct form of the momentum equation for situations in which the inertia is not constant, we need to adopt a Lagrangian description with generalized coordinates q_i. The Langevin equation with varying inertia $m(q)$ and varying friction $\zeta(q)$ is then

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(m_{ij}\dot{q}_j\right) - \tfrac{1}{2}\frac{\partial m_{jk}}{\partial q_i}\dot{q}_j\dot{q}_k + \zeta_{ij}\dot{q}_j \;=\; f_i^s + f_i^r\,.$$

Now the typical Brownian displacements are always much smaller than the scale over which the inertia and friction vary. We may therefore expand in the amplitude of the small Brownian motion

$$q_i(t) \;=\; q_i^{(0)} + q_i^{(1)}(t) + q_i^{(2)}(t) + \cdots$$

where $q^{(1)}(t)$ represents the standard linear Brownian motion together with the systematic motion from the steady forces obtained by using the constant values of the inertia $m(q^{(0)})$ and friction $\zeta(q^{(0)})$, while $q^{(2)}(t)$ is the small nonlinear correction which arises from the small changes of the inertia and friction between $q^{(0)}$ and $q(t)$. At second order one finds

$$m_{ij}\ddot{q}_j^{(2)} + q_k^{(1)}\frac{\partial m_{ij}}{\partial q_k}\ddot{q}_j^{(1)} + \dot{q}_k^{(1)}\frac{\partial m_{ij}}{\partial q_k}\dot{q}_j^{(1)} - \tfrac{1}{2}\frac{\partial m_{jk}}{\partial q_i}\dot{q}_j^{(1)}\dot{q}_k^{(1)} + \zeta_{ij}\dot{q}_j^{(2)} + q_k^{(1)}\frac{\partial \zeta_{ij}}{\partial q_k}\dot{q}_j^{(1)} \;=\; 0$$

where m, $\partial m/\partial q$, ζ and $\partial\zeta/\partial q$ are all evaluated at $q^{(0)}$. The equation shows that after a few inertia-friction relaxation times, i.e. on a time-scale much shorter than that required for the particle to diffuse a significant distance, a mean velocity is established. Denoting an average by an overbar, we find

$$\overline{\dot{q}_i^{(2)}} \;=\; \zeta_{ij}^{-1}\left(-\frac{\partial m_{jk}}{\partial q_l}\overline{q_l^{(1)}\ddot{q}_k^{(1)} + \dot{q}_l^{(1)}\dot{q}_k^{(1)}} + \tfrac{1}{2}\frac{\partial m_{kl}}{\partial q_j}\overline{\dot{q}_k^{(1)}\dot{q}_l^{(1)}} - \frac{\partial \zeta_{jk}}{\partial q_l}\overline{\dot{q}_l^{(1)}\dot{q}_k^{(1)}}\right)\,.$$

Using the definition of temperature $\overline{\dot{q}_i^{(1)}\dot{q}_j^{(1)}} = kTm_{ij}^{-1}(q^{(0)})$ and the expression for the diffusivity $\overline{q_i^{(1)}\dot{q}_j^{(1)}} = kT\zeta_{ij}^{-1}(q^{(0)})$, a constant independent of time, this can be reduced to

$$\overline{\dot{q}_i^{(2)}} \;=\; \zeta_{ij}^{-1}\frac{\partial}{\partial q_j}\left(kT\ln\sqrt{\det m}\right) + \frac{\partial D_{ij}}{\partial q_j}\,.$$

This mean velocity has two parts, one from the variations in the inertia and one from the variations in the diffusivity. The latter is the same as obtained in §3.3. The first is the same as the steady velocity which would result from the application of a pseudo potential force with a potential $-kT\ln\sqrt{\det m}$. Substituting the above mean drift into the dispersion equation which governs the probability distribution of configurations $P(q,t)$ yields, after a little rearrangement, a diffusion equation in the form (Kirkwood 1949)

$$\frac{\partial}{\partial t}\left(\frac{P}{\sqrt{\det m}}\right) + \frac{1}{\sqrt{\det m}}\frac{\partial}{\partial q_i}(\dot{q}_i^s P) = \frac{1}{\sqrt{\det m}}\frac{\partial}{\partial q_i}\left(\sqrt{\det m}D_{ij}\frac{\partial}{\partial q_j}\left(\frac{P}{\sqrt{\det m}}\right)\right).$$

In the absence of any systematic velocity $\dot{q}^s$ from externally applied steady forces, the equilibrium probability distribution varies with the changing inertia according to

$$P \propto \sqrt{\det m}.$$

Thus the equilibrium distribution is uniform only for those generalized coordinates which have $\sqrt{\det m}$ constant. We note that, because of the normalization, a non-uniform distribution would not change if the inertia was multiplied by a constant. Thus any non-uniformity would be preserved in the limit $m \to 0$. Moreover it would be possible to perform the Langevin simulations with a higher, and therefore more economical, value of the inertia to obtain a probability distribution identical to that for the real and often extremely small value.

4.3. *Corrected algorithm*

From the above investigation of the nonlinear effects of inertia which occur on the inertia-friction relaxation time-scale, we can now correct the earlier diffusion algorithms for simulated Brownian walks. To either Ermak & McCammon's algorithm or the mid-point algorithm, which both treated the variations in diffusivity correctly, one needs to add the extra mean Brownian velocity from the changes in inertia

$$+\zeta_{ij}^{-1}\frac{\partial}{\partial q_j}\left(kT\ln\sqrt{\det m}\right)$$

as an additional steady velocity. Equivalently one can apply the corrective pseudo-potential force with the potential $-kT\ln\sqrt{\det m}$. We note that the corrective force would not change if the inertia were multiplied by a constant, and so would not disappear in the limit $m \to 0$.

4.4. *A simple example*

We illustrate the problem caused by variations in inertia with a simple one-dimensional example. We consider a particle with Cartesian position $x(t)$ diffusing on the unit interval $[0, 1]$ with periodic end conditions and with a constant mass m and a constant friction coefficient ζ in the Cartesian description.

We make a coordinate transformation to a generalized coordinate

$$q(x) = \tfrac{1}{2} + (1/\sqrt{2})\sin\tfrac{1}{2}\pi(x-\tfrac{1}{2}).$$

As x increases from 0 to 1 so does q, except it does so non-uniformly. Now the kinetic energy is

$$\tfrac{1}{2}m\dot{x}^2 = \tfrac{1}{2}M\dot{q}^2 \quad \text{with} \quad M(q) = m\left(\frac{\mathrm{d}x}{\mathrm{d}q}\right)^2.$$

Similarly the rate of dissipation of energy against friction is

$$\zeta\dot{x}^2 = Z\dot{q}^2 \quad \text{with} \quad Z(q) = \zeta\left(\frac{\mathrm{d}x}{\mathrm{d}q}\right)^2.$$

For the generalized coordinate q, M is the mass and Z is the friction coefficient, both of which vary with position. When using the generalized coordinate, the diffusivity is $kT/Z(q)$.

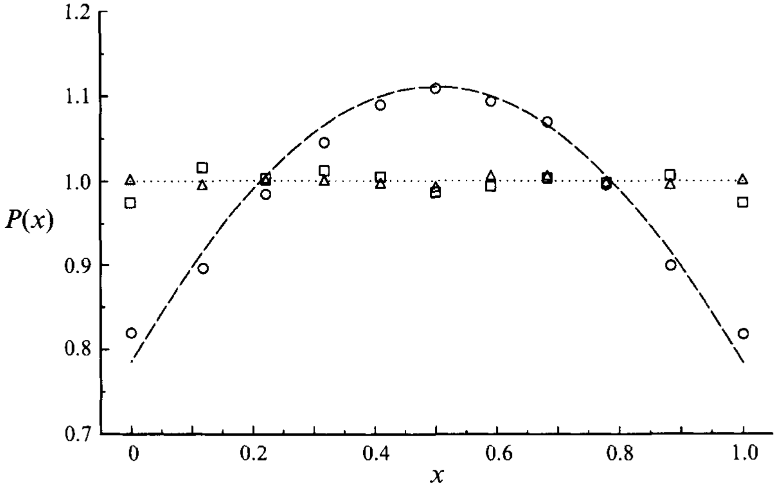

FIGURE 6. Variable inertia. The probability distributions $P(x)$ from various algorithms for a particle diffusing on $0 \leqslant x \leqslant 1$ using the generalized coordinate $q = \frac{1}{2} + (1/\sqrt{2}) \sin \frac{1}{2}\pi(x - \frac{1}{2})$. All the simulations have a time-step of $\delta t = 6.67 \times 10^{-5}$ and average over a time 10^3. The mid-point diffusion algorithm of §3.4 (○). The dashed curve gives the distribution which is uniform in q. The Langevin algorithm with Cartesian mass $m = 0.05$ (□). The mid-point diffusion algorithm with the correction potential force for the varying inertia (△).

Figure 6 gives the probability distribution for the position of the particle plotted as a function of the Cartesian position x. Although the simulations were all made in terms of the generalized coordinate q, at each time-step the x-position was calculated from the inverse coordinate transformation, and this x-position was used to construct the probability distribution in 10 equal-width bins in $0 \leqslant x \leqslant 1$. All the simulations in this §4 used a time-step of 6.67×10^{-5} and made averages over a time 10^3 so that the accuracy should be about 1%.

The points ○ in figure 6 give the results from the mid-point diffusion algorithm of §3.4, an algorithm which handles the varying diffusivity correctly. This algorithm produces a probability distribution which is erroneously *uniform in q* rather than correctly *uniform in x*. The dashed curve in figure 6 gives the probability distribution density function which is uniform in q, i.e. $P(x) = P(q)\mathrm{d}q/\mathrm{d}x = \mathrm{d}q/\mathrm{d}x$.

The result of the diffusion algorithms of §3 is wrong because the mass $M(q)$ varies. If we take this into account, we obtain the correct answer. The points □ in figure 6 are for a simulation of the Langevin equation in terms of the generalized coordinate q using the varying mass $M(q)$ (Cartesian mass $m = 0.05$) and the varying friction $Z(q)$. This simulation gives the correct probability distribution, uniform in x. We note that uniform in x is equivalent to an equilibrium probability density function for the generalized coordinate q with $P(q) \propto \sqrt{\det M(q)} \propto |\mathrm{d}x/\mathrm{d}q|$.

Finally we modify the diffusive algorithm for the generalized coordinate q by the application of the pseudo-potential force

$$+\frac{\partial}{\partial q}\left(kT \ln \sqrt{\det M}\right) .$$

With this modification, the diffusive algorithm gives the correct uniform distribution in x, as seen with the points △ in figure 6.

5. Stiff bonds versus rigid constraints

5.1. *The problem*

Often, complex Brownian systems, such as macromolecules, consist of chains of particles joined by bonds of very stiff springs. An accurate resolution of the rapid motion of these stiff bonds can require very small time-steps, for reasons of accuracy (§2.4) and numerical stability (§2.5). In order to concentrate on the more interesting changes of configuration, e.g. the coiling and uncoiling of the chain, one would like to replace the stiff bonds with rigid constraints, which would permit large time-steps.

Unfortunately the Brownian motion of the configurations of a system with rigid constraints can be different from that with stiff bonds. We have seen in §4.2 that in the absence of steady forces the equilibrium probability distribution varies with changes of inertia according to $P \propto \sqrt{\det m}$. In general the inertia of the rigid system m_r and that of the stiff system m_{sq} will vary differently with the configuration. Moreover if the stiffness of the springs varies with configuration, the springs themselves will produce steady forces which lead to an additional variation in the probability distribution, a problem we postpone to the end of this subsection.

Now the non-uniform probability distribution $P \propto \sqrt{\det m}$ was found in §4.2 to be the result of a contribution to the mean Brownian velocity

$$\zeta_{ij}^{-1}\frac{\partial}{\partial q_j}\left(kT\ln\sqrt{\det m}\right) .$$

We can therefore make a rigid Langevin system behave like a very stiff one, as far as configurational changes are concerned, by adding a mean velocity which cancels that of the rigid system and substitutes that of the stiff system, i.e. we need to add a corrective steady velocity

$$-\zeta_{ij}^{-1}\frac{\partial}{\partial q_j}\left(kT\ln\sqrt{\det m_r}\right) + \zeta_{ij}^{-1}\frac{\partial}{\partial q_j}\left(kT\ln\sqrt{\det m_{sq}}\right)$$

or equivalently apply a corrective potential force with potential

$$kT\ln\sqrt{\det m_r/\det m_{sq}} .$$

An alternative expression will be given in §6.1 which makes explicit the rigid constraints. Note that in the above we have assumed that the stiffness of the spring is independent of the generalized coordinates used to describe the rigid system.

Turning from the Langevin description to the diffusion description, the mid-point and Ermak & McCammon algorithms need to be modified according to §4.3 by including the additional Brownian velocity

$$+\zeta_{ij}^{-1}\frac{\partial}{\partial q_j}\left(kT\ln\sqrt{\det m_r}\right) ,$$

that is in order to obtain the correct results for the rigid system. On the other hand to obtain the correct results for the stiff system when applying these diffusion algorithms to the rigid system, we need to include instead the additional Brownian velocity

$$+\zeta_{ij}^{-1}\frac{\partial}{\partial q_j}\left(kT\ln\sqrt{\det m_{sq}}\right) .$$

As will be discussed in §6.1, this correction is modified when the diffusion algorithm for the rigid system is formulated in terms of Cartesian coordinates with tension forces to satisfy the constraints.

In the above we have viewed the inertia of the stiff system m_{sq} as a function of the generalized coordinates q used to describe the rigid system. Now in the underlying Cartesian description, the description which uses the Cartesian positions of each of the particles, the stiff system will have a mass m_{sx} which is constant independent of the configuration. To evaluate $\det m_{sq}$, we first need to extend the generalized coordinates q to Q so as to include also the stiff degrees of freedom. We make this extension using locally Cartesian coordinates orthogonal to the rigid subspace. The kinetic energy is then

$$\tfrac{1}{2}(m_{sx})_{ij}\dot{x}_i\dot{x}_j \;=\; \tfrac{1}{2}(m_{sq})_{kl}\dot{Q}_k\dot{Q}_l \quad\text{with}\quad (m_{sq})_{kl}(q) \;=\; (m_{sx})_{ij}\frac{\partial x_i}{\partial Q_k}\frac{\partial x_j}{\partial Q_l}\,.$$

It then follows that

$$\det m_{sq}(q) \;=\; A^2(q)\det m_{sx}$$

where the Jacobian $A = \partial(x)/\partial(Q)$ is just the generalized area element of the rigid system seen in the underlying Cartesian coordinates (because the extension from q to Q was taken to be locally Cartesian). A simple example will be given in §5.2.

While the stiffness of the springs does not normally depend on the configuration, there can be an apparent dependence when the spring laws are re-expressed in terms of the locally Cartesian stiff coordinates used to extend q to Q. This does occur when the direction of the spring force is not orthogonal to the constraint surface and its inclination changes around the surface (Helfand 1979). If the potential energy of the stiff springs is

$$\Phi \;=\; \tfrac{1}{2}\sum_{\text{stiff } ij} V_{ij}(q)Q_iQ_j \quad ,$$

then there will be components of $-\nabla\Phi$ (through the dependence $V_{ij}(q)$) along the rigid surface when one is not exactly on the constraint surface $Q_i = 0$ (stiff i). Thus a variation in the width of the potential well leads to a variation of the probability distribution around the constraint surface $\propto 1/\sqrt{\det V}$ (van Kampen 1981; van Kampen & Lodder 1984). A rigid system can be made to have this variation by including an additional steady Brownian velocity

$$-\zeta_{ij}^{-1}\frac{\partial}{\partial q_j}\left(kT\ln\sqrt{\det V}\right)\,.$$

Note that with different choices of V one rigid system can be made to behave like many different very stiff systems.

5.2. *A simple example*

We illustrate the problem of replacing stiff bonds by rigid constraints with a simple artificial example, discussed earlier by van Kampen & Lodder (1984). We consider a Brownian particle moving in two dimensions near the ellipse

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1\,.$$

All the simulations have $a = 2$ and $b = 1$. For the rigid system, the particle moves along the ellipse. For the stiff system, the particle is attracted towards the ellipse by a spring force $\boldsymbol{f}^s$. We illustrate two spring laws, the first acting along the line between the particle and the nearest point on the ellipse with a magnitude equal to this shortest distance multiplied by a spring constant κ, and the second given by the potential energy $\kappa ab(x^2/a^2 + y^2/b^2 - 1)^2/8$.

To ensure that the inertia and the friction of the rigid system change with the configuration, we give the unconstrained Cartesian particle an anisotropic but constant tensor mass and tensor friction. The equations governing the stiff system are then

$$\begin{aligned} m_x\ddot{x} + \zeta_x\dot{x} &= f_x^s + f_x^r(t) , \\ m_y\ddot{y} + \zeta_y\dot{y} &= f_y^s + f_y^r(t) . \end{aligned}$$

The simulations use $m_x = 0.1$, $m_y = 0.05$, $\zeta_x = 1$ and $\zeta_y = 2$, except where noted otherwise.

To describe the motion of the rigid system of the particle moving around the ellipse, we use the pseudo-angle θ in the expression for the position of the particle on the ellipse $\boldsymbol{x} = (a\cos\theta, b\sin\theta)$. The arclength position s is then given by

$$s(\theta) = \int_0^\theta A(\theta)\,\mathrm{d}\theta \quad \text{where} \quad A(\theta) = (a^2\sin^2\theta + b^2\cos^2\theta)^{1/2} .$$

Here $A(\theta)$ has the same meaning as in §5.1, i.e. it is the area element of the rigid system using the generalized coordinate θ as seen in the underlying Cartesian coordinates.

As the particle moves around the ellipse, the mass and the friction vary according to

$$M(\theta) = m_x a^2\sin^2\theta + m_y b^2\cos^2\theta \quad \text{and} \quad Z(\theta) = \zeta_x a^2\sin^2\theta + \zeta_y b^2\cos^2\theta .$$

Now in the Cartesian description, the stiff system has constant inertia $m_{sx} = \mathrm{diag}(m_x, m_y)$. To describe the stiff system using the generalized coordinate θ, we add the locally orthogonal measure of the distance from the ellipse. In the neighbourhood of the ellipse, the inertia for the stiff system then takes the form

$$m_{s\theta} = \begin{pmatrix} m_x a^2\sin^2\theta + m_y b^2\cos^2\theta & (m_y - m_x)ab\sin\theta\cos\theta/A \\ (m_y - m_x)ab\sin\theta\cos\theta/A & (m_x b^2\cos^2\theta + m_y a^2\sin^2\theta)/A^2 \end{pmatrix}$$

from which we can readily deduce $\det m_{s\theta} = A^2(\theta)\det m_{sx}$.

5.3. *Results*

Figure 7 gives the results of a number of simulations. The distributions are given as density functions with respect to arclength s. These were calculated by recording at the end of each time-step the presence of the particle, reflected onto the first quadrant, in one of 10 equally spaced bins in the angle $\theta = \tan^{-1}|ay/bx|$. The probability distributions are then converted to density functions of arclength by dividing the fraction of configurations in each bin by the arclength width of the bin.

The • points are for the basic diffusion algorithm applied to the first stiff system (force proportional to shortest distance to ellipse) with a spring stiffness $\kappa = 100$. It is necessary to have such a stiff spring in order to confine the particle within a small region, typically 0.1, of the ellipse. With such a stiff spring, numerical accuracy and stability requires a very small time-step $\delta t = 10^{-4}$. The ○ points are for a simulation of the Langevin equation with the stiff spring $\kappa = 100$ and mass $m_x = 0.01$ and $m_y = 0.005$. It is necessary to have a very small mass for this particular simulation with a stiff spring in order to keep the system overdamped. The very small mass in its turn requires an even smaller time-step $\delta t = 2.5 \times 10^{-5}$ for numerical stability and accuracy. Both types of simulation of the first stiff system produce a probability distribution which is uniform in arclength.

Changing to the second spring law (potential energy $\kappa ab(x^2/a^2 + y^2/b^2 - 1)^2/8$) with otherwise the same parameters produces the ■ results for the basic diffusion

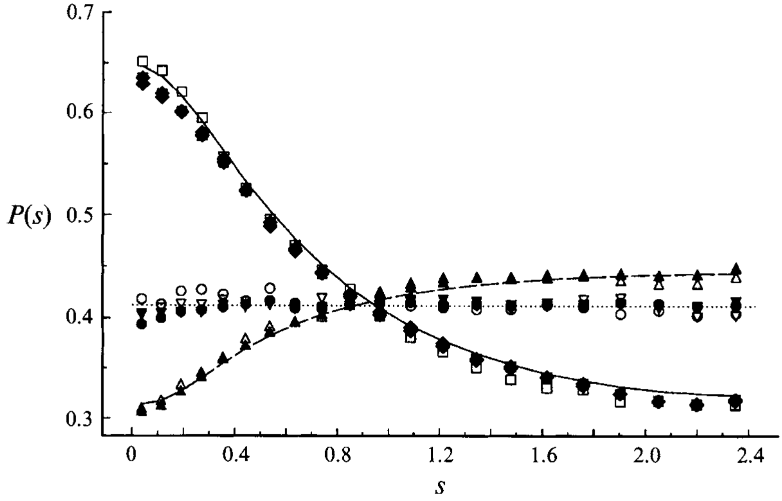

FIGURE 7. Stiff bonds versus rigid constraints. The probability distribution as a function of arclength $P(s)$ from various algorithms for a particle diffusing around an ellipse $x^2/a^2+y^2/b^2=1$ with $a=2$, $b=1$, $\zeta_x=1$, $\zeta_y=2$ and $kT=1$. The solid points are for diffusion algorithms while the open points are for Langevin simulations. The first stiff system with a spring constant $\kappa=100$ for diffusion with $\delta t=10^{-4}$ (●), and for the Langevin algorithm with mass $m_x=0.01$, $m_y=0.005$ and $\delta t=2.5\times10^{-5}$ (○). Using the same parameters, the results for the second stiff system (■, □). The solid curve gives the distribution uniform in θ. The dashed curve gives the theoretical results for the rigid system $P\propto\sqrt{M(\theta)}/A(\theta)$. All the rigid simulations use a time-step $\delta t=10^{-3}$. The Langevin algorithm for the rigid system (△). The mid-point diffusion algorithm including a correction force for the variations in inertia (▲). Adding a correction force to convert the rigid behaviour to that of a very stiff system, the first stiff system (▼, ▽) and the second stiff system (◆, ◇).

algorithm and the □ results for the Langevin algorithm. Now in the neighbourhood of the ellipse, the second spring force produces a force perpendicular to the ellipse with a magnitude $-\kappa ab(x^2/a^4+y^2/b^4)$ times the distance from the ellipse (found by substituting $\boldsymbol{x}=\boldsymbol{x}_0+\boldsymbol{n}h$ into $-\nabla\Phi$ with $\boldsymbol{x}_0$ on the ellipse, $\boldsymbol{n}$ the unit normal and h small). Thus the width of the potential well varies around the ellipse like $(x^2/a^4+y^2/b^4)^{-1/2}\propto 1/A(\theta)$. Hence the probability distribution will be like $1/A(\theta)$ in arclength, i.e. be uniform in θ as given by the solid curve in figure 7.

It is clear from the very small time-steps required to simulate the stiff system that it would be advantageous to replace the stiff bond with the rigid constraint that the particle moves only along the ellipse. In the rigid system, the equilibrium probability distribution changes from being uniform in arclength to a density function with respect to angle which is proportional to $\sqrt{M(\theta)}$, i.e. a density function with respect to arclength which is proportional to $\sqrt{M(\theta)}/A(\theta)$, which is denoted by the dashed curve in figure 7. Simulation of the Langevin equation using the generalized coordinate θ finds this distribution as shown by the Δ points in figure 7. Here a time-step of only $\delta t=10^{-3}$ was needed.

For diffusive simulations of the rigid system, the mid-point algorithm of §3.4 was used (because the diffusivity changes with configuration) and the additional velocity $+Z^{-1}\partial(kT\ln\sqrt{M(\theta)})/\partial\theta$ of §4.3 was included (because the inertia changes with the configuration). This diffusion simulation yields the correct distribution for the rigid system as displayed by the ▲ points. Again a time-step of only $\delta t=10^{-3}$ was needed.

To make the rigid system behave like the first very stiff system (with stiffness independent of configuration), we have to apply an additional corrective potential force

$$-\frac{\partial}{\partial\theta}\left(kT\ln\left(\sqrt{M(\theta)}/A(\theta)\right)\right).$$

The results of this modification are given by the ∇ points for the Langevin simulation and by the $\blacktriangledown$ points for the diffusion simulation (which has the net extra potential force $+\partial(kT\ln A(\theta))/\partial\theta$). The corrective potential force does indeed produce the predicted uniform distribution in arclength as for the first stiff system.

To make the rigid system behave like the second very stiff system (with the stiffness varying around the ellipse), we have to apply an additional corrective potential force

$$-\frac{\partial}{\partial\theta}\left(kT\ln\sqrt{M(\theta)}\right).$$

The results of this additional modification are given by the $\diamond$ points for the Langevin simulation, and by the $\blacklozenge$ points for the diffusion simulation (which has no net extra potential force). The corrective potential force does indeed produce the predicted uniform distribution in θ as for the second stiff system.

6. Cartesian coordinates and constraint forces

6.1. *The problem*

Instead of using generalized coordinates to describe a system with some rigid constraints it is often easier to use the Cartesian coordinates of the particles. These Cartesian coordinates must obey certain constraint conditions. To impose these conditions some constraint forces (Lagrange multipliers) must be applied. Two questions arise in using such a description to simulate Brownian motion. First, we need to know how to generate the appropriate random forces. Second, we need to be able to calculate the corrective force to make the rigid system behave instead like one with very stiff bonds. These questions were addressed by Hinch (1994) and we shall now recall the key results.

We consider a system, such as a polymer chain, governed by the equation

$$m_{ij}\ddot{x}_j + \zeta_{ij}\dot{x}_j + T^a\frac{\partial g^a}{\partial x_i} \;=\; f_i^s + f_i^r\,.$$

Here x_i is the $3N$ vector of the Cartesian coordinates of the N particles, m_{ij} and ζ_{ij} are $3N\times 3N$ diagonal tensors made up of the different masses and friction coefficients of the particles and T^a are the tensions maintaining K constraints in the form $g^a(x_i)=0$ for $a=1,2,\ldots,K$. Maintaining the constraints in time places restrictions on the velocity and acceleration

$$\dot{x}_i\frac{\partial g^a}{\partial x_i} \;=\; 0 \quad\text{and}\quad \ddot{x}_i\frac{\partial g^a}{\partial x_i} + \dot{x}_i\dot{x}_j\frac{\partial^2 g^a}{\partial x_i\partial x_j} \;=\; 0\,.$$

By taking different constraint functions $g^a(x_i)$ to express the same rigid system, it is possible to model different very stiff systems, i.e. stiff systems with different variations of the width of the potential well around the constraint surface. The potential energy of the corresponding stiff system is proportional to $\sum_a (g^a)^2$.

Now without the constraints the magnitude of the random thermal forces f^{ru} is given by the fluctuation-dissipation theorem

$$\overline{f_i^{ru}(t_1)f_j^{ru}(t_2)} = 2kT\zeta_{ij}\delta(t_1 - t_2)\,.$$

Using the earlier numerical representation of the delta-function, one generates in a simulation random forces at each time-step

$$f_i^{ru} = (24kT\zeta_{ii}/\delta t)^{1/2}\, r_i, \qquad i = 1, \ldots, N,$$

where we have used the fact that ζ is diagonal. When the constraints are applied, the random forces change and satisfy a fluctuation-dissipation theorem with that part of the friction tensor projected on to the constrained surface, denoted by $\zeta_{ij}^{\perp}$ in Hinch (1994). These constrained random forces f^{rc} are most easily constructed by firstly generating the unconstrained random forces f_i^{ru} as described above. Then one projects these forces to be orthogonal to the constraints $\partial g^a/\partial x_i$ by

$$f_i^{rc} = f_i^{ru} - T^{ra}\frac{\partial g^a}{\partial x_i}$$

with the tensions T^{ra} satisfying

$$\frac{\partial g^a}{\partial x_i}\frac{\partial g^b}{\partial x_i}T^{rb} \;=\; f_i^{ru}\frac{\partial g^a}{\partial x_i}\,.$$

It turns out that when using the Langevin description this projection of the random forces is unnecessary, but it is essential in the diffusion algorithms (Hinch 1994).

To convert the Brownian motion of the rigid system to that of an equivalent very stiff system, that which has potential energy proportional to $\sum_a (g^a)^2$, one must add a correction to the mean velocity which can be generated by applying a corrective potential force

$$-\frac{\partial}{\partial \boldsymbol{x}_i}\left(kT\ln\sqrt{\det}\right)\,.$$

For the Langevin simulations

$$\det = \det\left(M^{-1}\right)^{ab} \quad \text{with} \quad \left(M^{-1}\right)^{ab} \;=\; \sum_i m_{ii}^{-1}\frac{\partial g^a}{\partial \boldsymbol{x}_i}\cdot\frac{\partial g^b}{\partial \boldsymbol{x}_i}\quad ,$$

while for diffusion simulations

$$\det = \det\left(G^{-1}\right)^{ab} \quad \text{with} \quad \left(G^{-1}\right)^{ab} \;=\; \sum_i \frac{\partial g^a}{\partial \boldsymbol{x}_i}\cdot\frac{\partial g^b}{\partial \boldsymbol{x}_i}\,.$$

In these expressions, the sum is over the N particles at $\boldsymbol{x}_i$ and the constraints are now expressed as $g^a(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N) = 0$ for $a = 1, 2, \ldots, K$. These expressions involving Cartesian coordinates with the explicit dependence on the constraints are somewhat simpler that those in terms of generalized coordinates in §5.1. We see in these expressions that a change of the constraint functions would lead to a change in the distribution along the constraint surface according to the width of the well for the potential $\sum_a (g^a)^2$, as described at the end of §5.1.

A practical problem in simulating constrained systems is the accumulation of small numerical errors in satisfying the constraints. After a period of time one can wander some way from the constraint surface, and so it is necessary from time to time to project back onto the constraint surface. First-order forward time-stepping diffusion schemes are in general a disaster, because at each time-step they make an error $O(\delta x^2) = O(\delta t)$, and so accumulate an $O(1)$ error in an $O(1)$ time. Hence it is necessary to use second-order-accurate time-stepping schemes like the mid-point rule. Of course the mid-point algorithm is essential for diffusion simulations because the friction varies with configuration, and the alternative of evaluating Ermak & McCammon's $\nabla\cdot\boldsymbol{D}$ would be distinctly awkward. For Langevin simulations it is necessary to correct the velocity as well as the position, although the problem is less severe because the accumulated error in satisfying the constraints in an $O(1)$ time is found to be only $O(\delta t/m^{1/2})$ by both the forward time-stepping and mid-point methods.

The variation of friction enters subtly through the dependence of the constraints on the configuration. The mid-point diffusion algorithm of §3.4 (with random forces held constant while the friction varies) therefore becomes the following. First one generates the unconstrained random force f^{ru}. The components parallel to the constraints $\partial g^a/\partial x_i$ are projected out to form the constrained random force f^{rc}. This constrained random force is held frozen during the two-part time-step. For the first half-step, one uses the constraints evaluated at the original configuration, and for the second whole-step they are evaluated at the half-step configuration. Before each part step, one finds the instantaneous value of the tensions which keep the velocities orthogonal to the constraints acting for the configuration of that part step.

6.2. *Application to the ellipse example*

The general formalism of §6.1 looks much simpler when applied to our artificial ellipse example. The form of the constraint $g=0$ must be chosen with care because of the implicit assumption that the rigid system is the limit of a very stiff system with potential energy g^2. In order to be equivalent to the first spring law of §5.2 with a potential well which had a constant width around the ellipse, we take

$$g(x,y) \;=\; \tfrac{1}{2}\left(\frac{x^2}{a^2}+\frac{y^2}{b^2}-1\right)\left(\frac{x^2}{a^4}+\frac{y^2}{b^4}\right)^{-1/2}.$$

On the surface $g=0$, the gradient of this constraint is the unit normal to the ellipse

$$\frac{\partial g}{\partial \boldsymbol{x}} \;=\; \boldsymbol{n} \;=\; (x/a^2, y/b^2)\left(\frac{x^2}{a^4}+\frac{y^2}{b^4}\right)^{-1/2}.$$

To maintain this constraint in time requires that the velocity is orthogonal to the normal to the curve

$$\frac{\dot{x}x}{a^2}+\frac{\dot{y}y}{b^2} \;=\; 0$$

and that the accelerations (in a Langevin description) satisfy

$$\frac{\ddot{x}x}{a^2}+\frac{\ddot{y}y}{b^2}+\frac{\dot{x}^2}{a^2}+\frac{\dot{y}^2}{b^2} \;=\; 0\,.$$

The governing equation becomes

$$\begin{aligned} m_x\ddot{x}+\zeta_x\dot{x}+T\frac{x}{a^2} &= f^s_x+f^r_x\,,\\ m_y\ddot{y}+\zeta_y\dot{y}+T\frac{y}{b^2} &= f^s_y+f^r_y\,. \end{aligned}$$

To simulate a random Brownian walk, one first generates the unconstrained random force

$$f^{ru}_x=(24kT\zeta_x/\delta t)^{1/2}r_x, \quad\text{and}\quad f^{ru}_y=(24kT\zeta_y/\delta t)^{1/2}r_y$$

with two independent random numbers r_x and r_y uniformly distributed on $[-0.5, 0.5]$. These random forces are then constrained to be along the ellipse

$$\boldsymbol{f}^{rc} = \boldsymbol{f}^{ru} - \boldsymbol{n}(\boldsymbol{n}\cdot\boldsymbol{f}^{ru}) .$$

For the Langevin simulations the tension are found to be

$$T = \left(\frac{\dot{x}^2}{a^2} + \frac{\dot{y}^2}{b^2} + \left(f_x^s + f_x^{rc} - \zeta_x \dot{x}\right)\frac{n_x}{m_x} + \left(f_y^s + f_y^{rc} - \zeta_y \dot{y}\right)\frac{n_y}{m_y}\right) \Big/ \left(\frac{n_x^2}{m_x} + \frac{n_y^2}{m_y}\right)$$

while for the diffusion simulations

$$T = \left(\frac{\left(f_x^s + f_x^{rc}\right) n_x}{\zeta_x} + \frac{\left(f_y^s + f_y^{rc}\right) n_y}{\zeta_y}\right) \Big/ \left(\frac{n_x^2}{\zeta_x} + \frac{n_y^2}{\zeta_y}\right) .$$

To make the rigid system behave like the first very stiff system one must apply a corrective potential force

$$-\nabla kT \ln\left[\left(\frac{x^2}{m_x a^4} + \frac{y^2}{m_y b^4}\right)^{1/2} \left(\frac{x^2}{a^4} + \frac{y^2}{b^4}\right)^{-1/2}\right]$$

to the Langevin simulations. We note that the argument of the logarithm above is just the $\sqrt{M(\theta)}/A(\theta)$ of §5.2. For the diffusion simulations, the corrective potential force vanishes in this particularly simple example. On the other hand to make the diffusion simulations behave like a rigid system (with mass) one would obviously have to apply a force opposite to that above.

During the simulations the particle wandered off the constraint surface. For the diffusion simulations, the particle was moved back radially onto the ellipse every unit time interval (i.e. every 1000 time-steps) by dividing x and y by $(x^2/a^2 + y^2/b^2)^{1/2}$. For the Langevin simulations, the particle was first repositioned as above and then the normal component of the velocity subtracted. This was necessarily more frequently, every 0.1 time interval (again every 1000 time-steps). As a result of these projections, the particle was always within 1% of the constraint surface for the simulations described below.

The results of the simulations are given as probability density functions of arclength. As in §5, the positions were actually recorded in terms of θ, and the fraction of positions in one bin was later divided by the arclength width of the bin.

Numerical simulations have been made with values of the parameters $a = 2$, $b = 1$, $m_x = 0.1$, $m_y = 0.05$, $\zeta_x = 1$, $\zeta_y = 2$, $kT = 1$, with a time-step $\delta t = 10^{-3}$ for the diffusion simulations and $\delta t = 10^{-4}$ for the Langevin simulations, and averaging over a time 10^4. Figure 8 gives results of several simulations. The mid-point diffusion algorithm using the constrained random forces f^{rc} produces a uniform probability distribution in arclength, as shown by the • points. Wrongly using instead the unconstrained random forces f^{ru} produces an erroneously non-uniform distribution shown by the ■ points. The Langevin simulations of the rigid system can use either the constrained (○ points) or the unconstrained (□ points) forces, and produces the probability distribution $P \propto \sqrt{M(\theta)}/A(\theta)$ which is given by the dashed curve. In fact the forward time-stepping method applied to the Langevin equation produces identical random walks for the constrained and unconstrained random forces, in this problem and in general. Applying the corrective potential force to the Langevin simulations produces a uniform distribution in arclength, as shown by the Δ points, which is the correct behaviour of the first very stiff system. In this simple example of

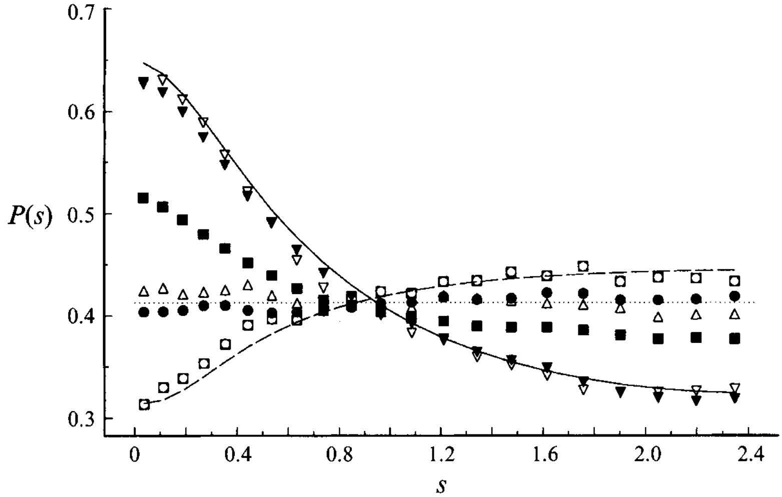

FIGURE 8. Cartesian coordinates and constraint forces. The probability distribution density function of arclength $P(s)$ from various algorithms for a particle diffusing around an ellipse $x^2/a^2+y^2/b^2 = 1$. The simulations have $a = 2$, $b = 1$, $\zeta_x = 1$, $\zeta_y = 2$, $m_x = 0.1$, $m_y = 0.05$, $kT = 1$, $\delta t = 10^{-3}$ for diffusion and $\delta t = 10^{-4}$ for the Langevin equation. The mid-point diffusion algorithm using constrained random forces (•) and using unconstrained random forces (■). The Langevin algorithm using constrained random forces (○) and using unconstrained random forces (□). The dashed curve is the theoretical result $P(s) \propto \sqrt{M(\theta)}/A(\theta)$. The Langevin algorithm including the correction force to convert the rigid behaviour to that of the first very stiff system (△). The diffusion algorithm (▼) and the Langevin algorithm (▽) including the correction force for the second stiff system.

an ellipse, the corrective potential force for the diffusion simulations vanishes, i.e. the mid-point algorithm gives the behaviour of the first very stiff system and not that of the rigid system.

The above discussion is for the first very stiff spring with a potential well of constant width. If we change constraint function to

$$g(x,y) = \frac{x^2}{a^2}+\frac{y^2}{b^2}-1$$

we can model the second very stiff spring of §5 with a potential well of varying width. This change in the constraint function does not affect the random part of the simulation, because changes in the length of the normal ∇g are accommodated entirely by a compensating reduction in the magnitude of the tension T. The correction potential forces are however modified, to

$$-\nabla kT \ln\left(\frac{x^2}{m_x a^4}+\frac{y^2}{m_y b^4}\right)^{1/2}$$

for the Langevin simulations, and to

$$-\nabla kT \ln\left(\frac{x^2}{a^4}+\frac{y^2}{b^4}\right)^{1/2}$$

for the diffusion simulations. The arguments of the logarithms above can be recognized as $\sqrt{M(\theta)}$ and $A(\theta)$ respectively from §5. Simulations with these correction potentials yield the results in figure 8 given by the ▼ and ▽ points respectively, results corresponding to the uniform distribution in θ (solid curve) of the second stiff system.

7. Application to a trimer

After the artificial examples in the previous sections, we now consider a more realistic system. The trimer molecule, consisting of a central monomer attached by two bonds to two outer monomers, is the simplest polymer molecule which can change its internal configuration when the individual links are rigid. We shall now apply to the trimer the ideas developed in the previous sections, and thereby this section will serve as a conclusion for the paper.

7.1. *Governing equations*

Let the central monomer be at $\boldsymbol{x}_2$ and the outer two at $\boldsymbol{x}_1$ and $\boldsymbol{x}_3$. We shall only use these Cartesian coordinates, and not use generalized coordinates for this example. Let the monomers have masses m_i and friction coefficients ζ_i. The governing equations are then

$$\begin{aligned} m_1\ddot{\boldsymbol{x}}_1 + \zeta_1\dot{\boldsymbol{x}}_1 &= \boldsymbol{f}_1^s + \boldsymbol{f}_1^b \qquad\quad + \boldsymbol{f}_1^r\,, \\ m_2\ddot{\boldsymbol{x}}_2 + \zeta_2\dot{\boldsymbol{x}}_2 &= \boldsymbol{f}_2^s - \boldsymbol{f}_1^b + \boldsymbol{f}_2^b + \boldsymbol{f}_2^r\,, \\ m_3\ddot{\boldsymbol{x}}_3 + \zeta_3\dot{\boldsymbol{x}}_3 &= \boldsymbol{f}_3^s \qquad\quad - \boldsymbol{f}_2^b + \boldsymbol{f}_3^r\,, \end{aligned}$$

where $\boldsymbol{f}^s$ is the steady force, $\boldsymbol{f}^b$ the bond connector force and $\boldsymbol{f}^r$ the random force. We consider only the case of freely hinged bonds, and so the bond forces $\boldsymbol{f}_i^b$ will be only in the direction of the bonds

$$\boldsymbol{d}_i = \boldsymbol{x}_{i+1} - \boldsymbol{x}_i, \qquad i = 1, 2\,.$$

For the stiff bonds, we take the linear spring law with a natural length l_i

$$\boldsymbol{f}_i^b = \kappa\left(|\boldsymbol{d}_i| - l_i\right)\boldsymbol{d}_i/|\boldsymbol{d}_i|\,.$$

This spring law only depends on the length of the bond and does not depend on the angle between the bonds. For the rigid system we take bond forces

$$\boldsymbol{f}_i^b = T_i\boldsymbol{d}_i, \qquad i = 1, 2$$

with two tensions T_i determined by the requirement of maintaining the two constraints

$$g^i(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) = \tfrac{1}{2}\left(\boldsymbol{d}_i^2 - l_i^2\right) = 0, \qquad i = 1, 2$$

in time, i.e. in the diffusion simulations

$$\dot{\boldsymbol{d}}_i \cdot \boldsymbol{d}_i = 0, \qquad i = 1, 2$$

and in the Langevin simulations

$$\ddot{\boldsymbol{d}}_i \cdot \boldsymbol{d}_i + \dot{\boldsymbol{d}}_i \cdot \dot{\boldsymbol{d}}_i = 0, \qquad i = 1, 2\,.$$

In both these cases, one obtains simultaneous linear equations for the two tensions.

To simulate a random Brownian walk, one first generates unconstrained random forces

$$\boldsymbol{f}_i^{ru} = (24kT\zeta_i/\delta t)^{1/2}\,\mathbf{r}_i, \qquad i = 1, 2$$

using random vectors $\mathbf{r}_i$ with components which are independent random numbers which have zero mean and variance $\frac{1}{12}$. The constrained random forces are then constructed by removing components in the direction of the constraints, i.e.

$$\boldsymbol{f}_1^{rc} = \boldsymbol{f}_1^{ru} + T_1^r\boldsymbol{d}_1, \qquad \boldsymbol{f}_2^{rc} = \boldsymbol{f}_2^{ru} - T_1^r\boldsymbol{d}_1 + T_2^r\boldsymbol{d}_2, \qquad \boldsymbol{f}_3^{rc} = \boldsymbol{f}_3^{ru} - T_2^r\boldsymbol{d}_2,$$

with T_i^r satisfying

$$\begin{pmatrix} 2d_1^2 & -\boldsymbol{d}_1 \cdot \boldsymbol{d}_2 \\ -\boldsymbol{d}_1 \cdot \boldsymbol{d}_2 & 2d_2^2 \end{pmatrix} \begin{pmatrix} T_1^r \\ T_2^r \end{pmatrix} = \begin{pmatrix} (\boldsymbol{f}_2^{ru} - \boldsymbol{f}_1^{ru}) \cdot \boldsymbol{d}_1 \\ (\boldsymbol{f}_3^{ru} - \boldsymbol{f}_2^{ru}) \cdot \boldsymbol{d}_2 \end{pmatrix} .$$

To convert the Brownian motion of the rigid system to that of the very stiff system, one applies a corrective potential force. In the case of Langevin simulations, the formalism of §6.1 can be shown (Hinch 1994) to reduce to

$$\begin{aligned} (\boldsymbol{f}_1^s, \boldsymbol{f}_2^s, \boldsymbol{f}_3^s) &= Q\,(\boldsymbol{x}_3 - \boldsymbol{x}_2, \boldsymbol{x}_2 - \boldsymbol{x}_3 + \boldsymbol{x}_2 - \boldsymbol{x}_1, \boldsymbol{x}_1 - \boldsymbol{x}_2) \\ \text{where} \quad Q &= \frac{kTm_1m_3\cos\theta}{l_1l_2\left((m_1+m_2)(m_2+m_3) - m_1m_3\cos^2\theta\right)} \end{aligned}$$

where θ is the included angle between the two bonds defined by

$$\cos\theta = -(\boldsymbol{d}_1 \cdot \boldsymbol{d}_2)/l_1l_2 .$$

In the case of diffusion simulations, the corrective potential force has the same form but with $m_1 = m_2 = m_3$.

The numerical simulations are for the quiescent trimer with no externally applied forces. We shall be interested in the probability distribution of the included angle between the two bonds, θ. Because of the axisymmetry in three dimensions which gives the probability density function for θ an intrinsic factor of $\sin\theta$, it is more convenient to consider the probability density function for $\cos\theta$, which is also easier to evaluate from the Cartesian positions of the monomers. If all configurations are equally probable, then $P(\cos\theta) = \frac{1}{2}$.

7.2. *Langevin simulations*

Figure 9 gives the results for Langevin simulations. All the simulations have unit bond lengths $l_1 = l_2 = 1$, friction coefficients $\zeta_1 = \zeta_3 = 1$ and $\zeta_2 = 0.25$, and unit temperature $kT = 1$. Sufficient accuracy could be obtained by averaging over a time 4000.

The ○ points are for stiff bonds. A spring constant of $\kappa = 100$ was necessary in order to keep the bonds within 10% of their natural length. To keep the system overdamped, the masses had to be small $m_1 = m_3 = 0.00025$ and $m_2 = 0.000125$. With these small masses, a time-step of $t = 2.5 \times 10^{-5}$ was necessary in order to satisfy the stability criterion of §2.5 that $\zeta\delta t/m < 1$. For Langevin simulations of a stiff spring, it is sufficient to use a simple forward time-stepping scheme. The results show that the included angle of this trimer is uniformly distributed.

The rigid system was simulated in terms of the Cartesian positions using constraint tensions. For the rigid systems it is possible to have masses larger than above, yet still smaller than unity. We used $m_1 = m_3 = 0.05$ and $m_2 = 0.025$, which permits the larger time-step $\delta t = 2 \times 10^{-4}$, and averaged over a time 10^4. Whenever the length of a bond deviated by more than 0.5%, the lengths of the bonds were normalized to unity. Similarly whenever a velocity developed a component exceeding 0.5% in the direction of the constraints, the components of the velocities parallel to the constraints were projected out. For Langevin simulations there is no advantage in using the mid-point method to help maintain the constraints, so we used forward time-stepping. As the random walk generated by a forward time-stepping method with the constrained random forces $\boldsymbol{f}^{rc}$ is identical to that for the unconstrained random forces $\boldsymbol{f}^{ru}$, we used the latter. The results of this simulation are shown by the □ points in figure 9. The probability distribution is not uniform in the included angle, but instead

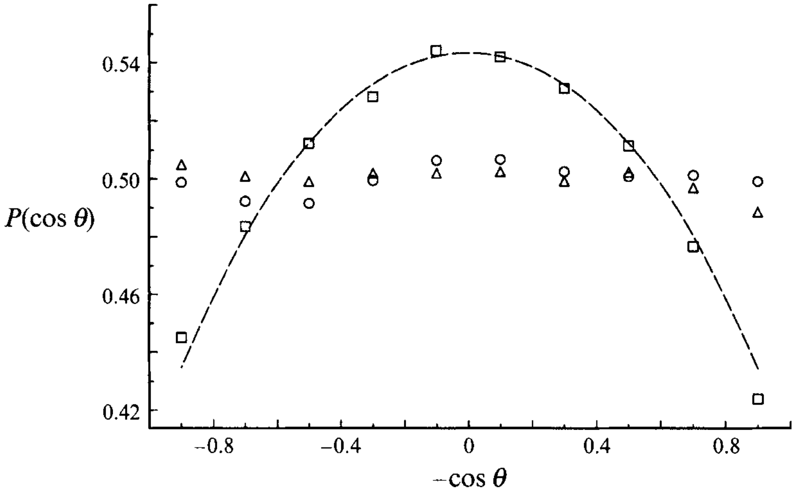

FIGURE 9. Langevin simulations of a trimer with $l_1 = l_2 = 1$, $\zeta_1 = \zeta_3 = 1$, $\zeta_2 = 0.25$, $kT = 1$. The probability distribution $P(\cos\theta)$ of the included angle θ. Stiff springs (○) with spring constant $\kappa = 100$, and with $m_1 = m_3 = 0.00025$, $m_2 = 0.000125$, $\delta t = 2.5 \times 10^{-5}$ and an averaging time 4×10^3. Rigid bonds (□) with $m_1 = m_3 = 0.05$, $m_2 = 0.025$, $\delta t = 2 \times 10^{-4}$ and an averaging time of 10^4. The dashed curve is the theoretical result for the rigid system $P \propto \sqrt{9 - 4\cos^2\theta}$. The rigid system including the correction force to convert the rigid behaviour to that of the very stiff system (△).

$P \propto \sqrt{\det M}$ gives

$$P(\cos\theta) \propto \sqrt{(m_1 + m_2)(m_2 + m_3) - m_1 m_3 \cos^2\theta}$$

which is given in figure 9 by the dashed curve.

The final Langevin simulation adds the corrective potential force to the above algorithm. The results are given by the △ points in figure 9. Applying the potential force does convert the behaviour of the rigid system into that of the stiff system by producing a uniform distribution in the included angle.

7.3. *Diffusion simulations*

Figure 10 gives the results for the diffusion simulations of the trimer. The bond lengths, friction coefficients and temperature are the same as for the Langevin simulations above with an averaging time of 4×10^3 in all cases. The ○ points are for stiff bonds with a spring constant $\kappa = 100$. The time-step need not be quite so small for this stiff spring in a diffusion simulation; $\delta t = 10^{-4}$ was found to be adequate. As the stiff system has constant friction and constant inertia, a simple forward time-stepping scheme can be used. The results show that the included angle is uniformly distributed for a stiff diffusive simulation.

Diffusive simulations of the rigid system are complicated by the need (i) to constrain the random forces (§6.1), and (ii) to use the mid-point algorithm (§3.4) because the constraints make the friction vary with the configuration. Moreover to keep control of the accumulation of the small numerical errors in satisfying the constraints, a mid-point algorithm is essential. Whenever the length of a bond deviated by more than 0.5%, the lengths of the bonds were normalized to unity. The □ points show the results of such simulations, with $\delta t = 10^{-3}$. These results should be contrasted

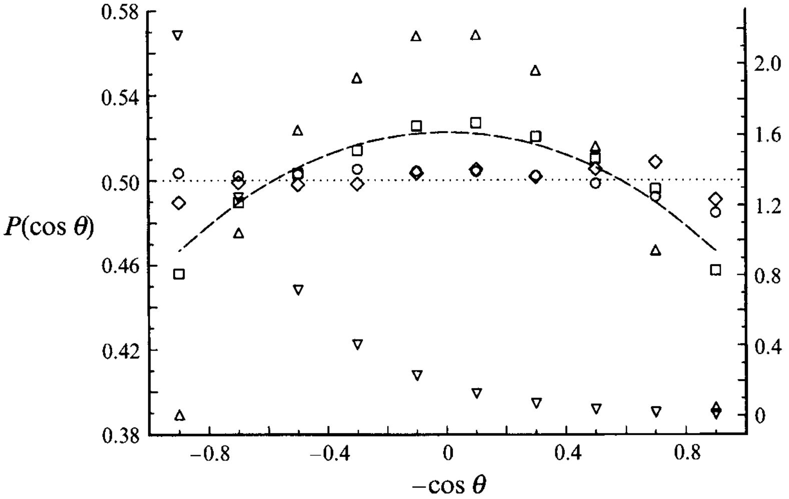

FIGURE 10. Diffusion algorithms for a trimer with $l_1 = l_2 = 1$, $\zeta_1 = \zeta_3 = 1$, $\zeta_2 = 0.25$, $kT = 1$. The probability distribution $P(\cos\theta)$ of the included angle θ. Stiff springs (○) with spring constant $\kappa = 100$, and with $\delta t = 10^{-4}$ and an averaging time of 4×10^4. Rigid bonds (□) using the mid-point rule and constrained random forces with $\delta t = 10^{-3}$ and an averaging time of 4×10^3. The dashed curve is the theoretical result for the rigid system with beads of equal mass $P \propto \sqrt{4 - \cos^2\theta}$. The rigid system using instead the unconstrained random forces (△) and using instead the forward time-stepping (▽) (right-hand axis for these data only). The rigid system including the correction force to convert the rigid behaviour to that of the very stiff system (◇).

with the △ points from erroneously using instead unconstrained random forces, and with the ▽ points from erroneously using instead a simple forward time-stepping algorithm. (In fact one can use the unconstrained random forces in the special case of the monomers all having identical friction coefficients.) Now it can be shown that, when using a Cartesian description with the constraint forces in a diffusion simulation, the equilibrium probability distribution is given by setting all the masses equal in the expression above for the distribution when there are non-zero masses, i.e. Kramers' (1946) distribution

$$P(\cos\theta) \propto \sqrt{1 - \tfrac{1}{4}\cos^2\theta}\,.$$

This distribution is plotted in figure 10 with the dashed curve.

Finally the ◇ points in figure 10 give the results of adding the corrective potential force to the diffusion simulation which used the mid-point algorithm and the constrained random forces. This simulation has the probability distribution of the very stiff system.

7.4. *Conclusions*

We conclude that it is possible to make economical diffusion simulations of a very stiff system by replacing the stiff bonds with rigid constraints, but care is needed: (*a*) the variation of the friction with configuration requires a mid-point algorithm to be used, with the random force held constant during the two part-steps while the friction varies; (*b*) the random forces must be chosen to satisfy the constraints; and (*c*) a corrective potential force must be applied to convert the rigid behaviour into that of the very stiff system. The first two complications do not arise if one uses a

Langevin simulation, although that is more expensive numerically because the small masses require a smaller time-step. We note that instead of the real very small masses, one could use artificially larger values to obtain the same results more economically, because the diffusion does not depend on the absolute value of the inertia only its fractional variations. The complication (*c*) does not arise if one retains the stiff springs, although such simulations can be extremely expensive.

P.S.G. was supported by the Hackett Studentship Awarded by the University of Western Australia. The computations were supported in part by the SERC CSI grant GR/H57585 and in part by the DTI LINK programme on colloids.

REFERENCES

Ermak, D. L. & McCammon, J. A. 1978 Brownian dynamics with hydrodynamic interaction. *J. Chem. Phys.* **69**, 1352–1360.

Fixman, M. 1978*a* Simulation of polymer dynamics. I. General theory. *J. Chem. Phys.* **69**, 1527–1537.

Fixman, M. 1978*b* Simulation of polymer dynamics. II. Relaxation rates and dynamic viscosity. *J. Chem. Phys.* **69**, 1539–1545.

Fixman, M. 1986 Construction of Langevin forces in simulations of hydrodynamic interactions. *Macromolecules* **19**, 1204–1207.

Hassager, O. 1974 Kinetic theory and rheology of bead-rod model for macromolecular solution. I. Equilibria & steady flow properties . *J. Chem. Phys.* **60**, 2111–2124.

Helfand, E. 1979 Flexible vs rigid constraints in statistical mechanics. *J. Chem. Phys.* **71**, 5000–5007.

Helfand, E. Wasserman, Z. R. & Weber, T. A. 1980 Brownian dynamics study of polymer conformational transitions. *Macromolecules* **13**, 526–533.

Heyes, D. M. & Melrose, J. R. 1993 Brownian dynamics simulations of model hard-sphere suspensions. *J. Non-Newtonian Fluid Mech.* **46**, 1–28.

Hinch, E. J. 1994 Brownian motion with stiff bonds and rigid constraints. *J. Fluid Mech.* **271**, 219–234.

Hinch, E. J. & Nitsche, L. C. 1993 Nonlinear drift interactions between fluctuating colloidal particles: oscillatory and stochastic motions. *J. Fluid Mech.* **256**, 343–401.

Kampen, N. G. van 1981 Statistical mechanics of trimers. *Applied Sci. Res.* **37**, 67–75.

Kampen, N. G. van & Lodder, J. J. 1984 Constraints. *Am. J. Phys.* **52**, 419–424.

Kirkwood, J. G. 1949 The statistical mechanical theory of irreversible processes in solutions of flexible macromolecules. Visco-elastic behaviour. *Rec. Trav. Chim.* **68**, 649–660.

Kramers, H. A. 1946 The behaviour of macromolecules in inhomogeneous flows. *J. Chem. Phys.* **14**, 415–424.

Kubo, R., Toda, M & Hasitsume, N. 1985 *Statistical Physics II* §1.6. Springer.

Luty, B. A., Wade, R. C., Madura, J. D., Davis, M. E., Briggs, J. A. & McCammon, J. A. 1993 Brownian dynamics simulations of diffusional encounters between triose phosphate isomerase and glyceraldehyde phosphate: electrostatic steering of glyceraldehyde phosphate. *J. Chem. Phys.* **97**, 233–237.

Nambi, P., Wierzbicki, A. & Allison, S. A. 1992 Intermolecular interaction between bovine pancreatic trypsin inhibitor molecules probed by Brownian dynamics simulations. *J. Chem. Phys.* **95**, 9595–9600.

Northrup, S. H. & Erickson, H. P. 1992 Kinetics of protein-protein association of diffusion-influenced biomolecular reactions. *Proc. Natl. Acad. Sci. USA* **89**, 3338–3342.

Øskendal, B. 1985 *Stochastic Differential Equations: An Introduction with Applications*. Springer-Verlag.

Parnas, R. S. & Cohen, Y. 1991 Response of a terminally anchored polymer chain to a simple shear flow. *Macromolecules* **24**, 4646–4656.

Pear, M. R. & Weiner, J. H. 1979 Brownian dynamics study of a polymer chain of linked rigid bodies. *J. Chem. Phys.* **71**, 212–224.

Rallison, J. M. 1979 The role of rigidity constraints in the rheology of dilute polymer solutions . *J. Fluid Mech.* **93**, 251–279.

REY, A., FREIRE, J. J. & GARCÍA DE LA TORRE, J. 1989 Brownian dynamics of a flexible polymer. Internal modes and quasielastic scattering function. *J. Chem. Phys.* **90**, 2035–2041.

RIGOS, A. A. & WILEMSKI, G. 1992 Brownian dynamics simulations of an order-disorder transition in sheared sterically stabilized colloid suspensions *J. Chem. Phys.* **96**, 3981–3986.

SHERWOOD, J. D. 1992 Brownian dynamics simulation of a 2-D suspension of charged colloidal plates under shear. *J. Non-Newtonian Fluid Mech.* **43**, 195–228.