Algorithms for Brownian dynamics simulation

A. C. Brańka
Institute of Molecular Physics, Polish Academy of Sciences, Smoluchowskiego 17/19, 60-179 Poznań, Poland

D. M. Heyes
Department of Chemistry, University of Surrey, Guildford, GU2 5XH, United Kingdom
(Received 26 January 1998)

Several Brownian dynamics numerical schemes for treating one-variable stochastic differential equations at the position of the Langevin level are analyzed from the point of view of their algorithmic efficiency. The algorithms are tested using a one-dimensional biharmonic Langevin oscillator process. Limitations in the conventional Brownian dynamics algorithm are shown and it is demonstrated that much better accuracy for dynamical quantities can be achieved with an algorithm based on the stochastic expansion (SE), which is superior to the stochastic second-order Runge-Kutta algorithm. For static properties the relative accuracies of the SE and Runge-Kutta algorithms depend on the property calculated. [S1063-651X(98)05708-0]

PACS number(s): 02.70.Ns, 02.50.Ey, 02.60.Cb, 05.20.−y

I. INTRODUCTION

An increasing number of problems in dispersed phase systems such as polymer solutions, colloidal liquids, and their mixtures are being modeled by computer simulation. In such systems the influence of the large number of solvent molecules on the polymer or colloidal particles is treated by a combination of random forces and frictional terms. In effect, the dispersed systems are described not by Newton's equations of motion but by a form of Langevin equation. Compared to the well-established molecular dynamics (MD) technique for solving Newton's equations, the stochastic dynamics (SD) algorithms for solving the Langevin equations are considerably less well developed because of their stochastic nature. The most simple form of SD, devised by Ermak, is called Brownian dynamics (BD), in which the stochastic force contains no correlations in space or time and the equations are solved with a low-order algorithm [1]. This algorithm is still widely used for BD simulations and has been applied to a variety of problems; see, for example, [2–5].

One of the main problems with this technique is that very small time steps are required; otherwise drifts in calculated quantities with increasing time step can occur even though the numerical procedure appears to be stable. (In this respect, it differs from MD where noticeable systematic drift in calculated properties and algorithm instability are closely linked.) Of course, eventually at a large enough time step even the Ermak BD algorithm becomes unstable. These obvious drawbacks follow from the fact that the BD method is at the level of the first-order Euler method for ordinary differential equations, which is well known to be a simple and straightforward but rather inaccurate procedure.

One possible way to improve the efficiency of the BD method (as measured by the magnitude of the time step that can be used with tolerable drift) is to search for an appropriate second- or higher-order algorithm. This worthy ambition, however, is not so simple to achieve in practice; only a few proposals have appeared in the literature and none of them can be considered to be fully satisfactory. Most of them are based on extending deterministic Runge-Kutta algorithms to include stochastic terms. Algorithms along this line have been proposed, e.g., by Helfand [6] and Iniesta and Garcia de la Torre [7], and recently one has been rigorously developed for the one-variable case by Honeycutt [8]. These methods require more than one evaluation of the particle force per time step, which clearly reduces its efficiency, but this can be compensated for by being able to employ larger time steps, typically several times larger than with BD. It has been argued, however, that such a method offers some advantages over conventional BD method, e.g., it gives more accurate results for the same amount of computer time [7]. The method has been applied to several macromolecular dispersions, including a DNA chain [9] and suspensions of charged rodlike colloidal particles [10].

van Gunsteren and Berendsen proposed a Verlet-type stochastic algorithm that is versatile enough to allow the performance of simulations covering a broad range of the friction coefficients [11]. It needs only one evaluation of the force per time step and reduces to a simple algorithmic form in the large friction coefficient limit. This algorithm has been used to model, for example, polymer dynamics in solution [12].

In the present note we shall consider, from the point of view of BD algorithms, the iterative solution of stochastic differential equations developed by Honeycutt [8]. On the basis of this one-variable solution mutual relations between different BD algorithms can be analyzed for the simplest case of one-dimensional Brownian motion in a potential in the diffusion limit. Such a simple model case should be considered before going to a multivariable case because some exact results are available for it, which facilitates verification and more rigorous comparisons between the different algorithms.

The basic BD algorithms are considered in Sec. II. In Sec. III a simple numerical test is discussed. Conclusions are in Sec. IV.

II. ALGORITHMS

The evolution of the position of a Brownian particle in a potential field U is described by the stochastic differential equation [13]

$$\frac{dx}{dt}=\beta DF+D^{1/2}\xi, \tag{1}$$

which corresponds to the equation for the probability distribution function $P(x,t)$,

$$\frac{\partial P}{\partial t}=D\,\frac{\partial}{\partial x}\left(\frac{\partial P}{\partial x}-\beta FP\right). \tag{2}$$

Here $\beta=1/k_BT$, where k_B is Boltzmann's constant, T is the temperature, $F=-dU/dx$ is the force acting on the particle, D is the diffusion constant, and ξ is the Gaussian white noise process. The above equations are a one-variable version of the position Langevin and the Smoluchowski equation, which form the basis for the Brownian dynamics method (without many-body hydrodynamic interactions).

The stationary solution of the Smoluchowski equation is the Boltzmann distribution

$$P(x,\infty)=\text{const}\times e^{-\beta U}. \tag{3}$$

From the Smoluchowski equation the explicit results for the mean square displacement of a particle position can be calculated up to fourth order in time [14],

$$\langle\Delta x^2\rangle=2Dt+\frac{D^2}{T}\langle F'\rangle t^2+\frac{D^3}{3T^2}\langle F'^2\rangle t^3 - \frac{D^4}{12T^3}[T\langle F''^2\rangle-\langle F'^3\rangle]t^4. \tag{4}$$

Here and in the following T means k_BT, $F'=dF/dx$, and $F''=d^2F/dx^2$. Some algorithms proposed for solving Eq. (1), which we will consider, have the following form. The conventional Brownian dynamics (CBD) algorithm

$$x(\Delta t)=x_0+\frac{D}{T}F\Delta t+\Gamma_0, \tag{5}$$

the van Gunsteren–Berendsen (GB) algorithm

$$x(\Delta t)=x_0+\frac{D}{2T}(2F+\dot{F}\Delta t)\Delta t+\Gamma_0, \tag{6}$$

and the second-order stochastic Runge-Kutta (SRK) algorithm

$$x(\Delta t)=x_0+\frac{D}{2T}(F_1+F_2)\Delta t+\Gamma_0, \tag{7}$$

where $F_1=F(x_0)$ and $F_2=F[x_0+(D/T)F\Delta t+\Gamma_0]$. The time derivative of the force in the GB algorithm is conventionally approximated by $\dot{F}=[F(t)-F(t-\Delta t)]/\Delta t$ and Γ_0 in the above equations is a random number sampled from a Gaussian distribution with zero mean and width $\langle\Gamma_0^2\rangle=2D\Delta t$.

For a general stochastic differential equation such as Eq. (1) the following expansion for $x(\Delta t)$ occurs [8]:

$$x(\Delta t)=x_0+\frac{D}{T}F\Delta t+\frac{D^2}{2T^2}FF'\Delta t^2+\Gamma_0+\frac{D}{T}F'\Gamma_1 + \frac{D}{2T}F''G, \tag{8}$$

where terms of an order higher than Δt^2 are omitted and

$$\Gamma_1=\int_0^{\Delta t}\Gamma_0 ds, \tag{9}$$

$$G=\int_0^{\Delta t}\Gamma_0^2 ds \tag{10}$$

are random numbers. The expansion of Eq. (8) results from an integration of Eq. (1), the Taylor expansion for F, and its repeated insertion into itself [8]. It is important to recognize that the random number Γ_1 is also, like Γ_0, a Gaussian random number with the properties

$$\langle\Gamma_1\rangle=0,\quad \langle\Gamma_1^2\rangle=\tfrac{2}{3}D\Delta t^3,\quad \langle\Gamma_1\Gamma_0\rangle=D\Delta t^2. \tag{11}$$

Thus the term in Eq. (8) involving Γ_1 is of order $\Delta t^{3/2}$. The last term is of order Δt^2, but its nonlinearity does not allow us to obtain a more explicit representation. In the following the algorithm based on the expansion of Eq. (8) will be called the stochastic expansion (SE) algorithm.

An important quantity enabling us to differentiate between the various algorithms is the mean-square displacement in a time step (MSD1). Its exact form follows from Eq. (4). The CBD algorithm gives only the trivial linear approximation

$$\langle\Delta x^2\rangle=2D\Delta t+\frac{D^2}{T}\langle F'\rangle\Delta t^2+\delta_{\text{CBD}}\Delta t^2 \tag{12}$$

and the error in the second order term is

$$\delta_{\text{CBD}}=\frac{D^2}{T}\left[\frac{1}{T}\langle F^2\rangle-\langle F'\rangle\right]. \tag{13}$$

For many physical realizations the second derivative of the interaction potential is predominantly positive, which implies $\delta_{\text{CBD}}>0$. Thus the CBD algorithm generally overestimates the MSD1.

The GB algorithm also yields an error in the second-order term,

$$\langle\Delta x^2\rangle=2D\Delta t+\frac{D^2}{T}\langle F'\rangle\Delta t^2+\delta_{\text{GB}}\Delta t^2, \tag{14}$$

where

$$\delta_{\text{GB}}=\delta_{\text{CBD}}+\frac{3}{2}\langle F^2\rangle\left(1-\frac{\langle F(\Delta t)F(0)\rangle}{\langle F^2\rangle}\right). \tag{15}$$

As the normalized autocorrelation function is less than unity, the second contribution in the δ_{GB} is *always* positive. This means, rather surprisingly, that in general the GB algorithm yields larger errors than the CBD algorithm.

The SRK algorithm gives

$$\langle\Delta x^2\rangle=2D\Delta t+\frac{D^2}{T}\langle F'\rangle\Delta t^2-\delta_{\text{SRK}}\Delta t^2, \tag{16}$$

with an error contribution

$$\delta_{\mathrm{SRK}}=\frac{D^3}{2T^3}\left(\langle F_1^2\rangle-\langle F_1F_2\rangle\right), \tag{17}$$

which is always positive and for small Δt can be approximated by $\langle F'F^2\rangle\Delta t/2$. Thus the SRK algorithm gives the correct second-order term for MSD1 and yields an underestimation of MSD1 with leading order Δt^3.

Equation (8) yields the following expression for the MSD1:

$$\langle\Delta x^2\rangle=2D\Delta t+\frac{D^2}{T}\langle F'\rangle\Delta t^2+\frac{D^2}{T}\left[\frac{1}{T}\langle F^2\rangle+\langle F'\rangle\right]\Delta t^2 + \frac{D^3}{T^2}\left[\frac{1}{T}\langle F'F^2\rangle+\frac{1}{2}\langle F''F\rangle+\frac{2}{3}\langle F'^2\rangle\right]\Delta t^3. \tag{18}$$

For many physical interactions (e.g., $U\sim x^{2n}$) the second term of order Δt^2 is equal to zero and the expansion of Eq. (8), like the SRK algorithm, gives the correct MSD1 with some deviation in terms of order Δt^3.

III. NUMERICAL CALCULATIONS

In order to compare the efficiency of the above algorithms with increasing Δt and to establish how the MSD1 errors influence the static and dynamic quantities we have considered the dynamics of a Brownian particle in an external potential $U=kx^4/4$, a potential for which $F''\neq 0$ and the stationary solution exists. In the calculations $k=1$, $T=1$, and $D=1$ were used and the averages were calculated from simulations of about 5×10^8 time steps. In the SE algorithm based on Eq. (8) the G term was approximated by $\frac{1}{2}\Delta t\Gamma_0^2$ and the two correlated random numbers Γ_0 and Γ_1 were sampled from a bivariate Gaussian distribution.

Perhaps the most important yet simplest dynamical quantity is the one we shall consider: the time-dependent mean square displacement. In many-particle systems the MSD in the long-time limit gives the self-diffusion coefficient of the particle. The general behavior of the MSD calculated with the various algorithms is shown in Fig. 1 for a large time step $\Delta t=0.05$ (for other time steps the results are qualitatively very similar). On the scale of the figure the MSD curves obtained from SRK and SE calculations coincide with the exact curve (i.e., the curve produced by all algorithms in the limit of very small time step). The CBD curve very slowly approaches from above the exact curve. Also the GB curve deviates considerably from the exact curve at short times but converges relatively quickly at longer times (in about $10-15\Delta t$) to the correct form. Notice that, in accordance with Eq. (15), it starts above the CBD curve. Therefore, all algorithms reproduce correctly the long-time behavior of the MSD, but differ considerably in their ability to reproduce its intermediate- and particularly short-time behavior. This is clearly seen in the enlargement in Figs. 1 and 2. In the inset a slight difference between the SRK and SE algorithms emerges. The SRK algorithm preserves the initial small MSD1 deviation for a long time. The SE method, in contrast, converges quickly to the exact curve. Figure 2 illustrates a

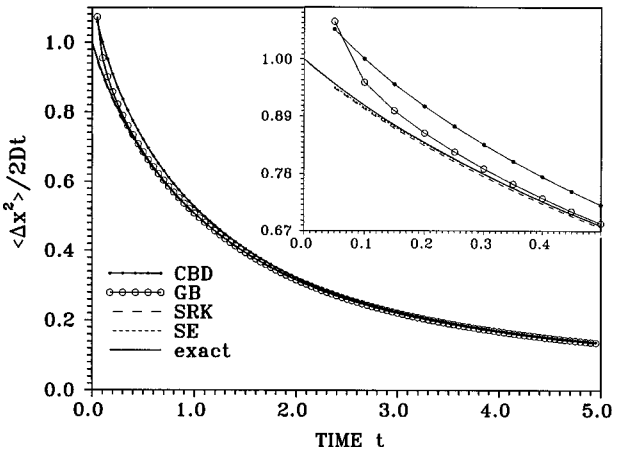

FIG. 1. Mean square displacement of a biharmonic Langevin oscillator obtained from the four BD algorithms discussed in the text with time step $\Delta t=0.05$. The curve labeled ''exact'' is the limit obtained from calculations with $\Delta t=0.001$. The inset presents an enlargement of the short- and intermediate-time regions.

significant influence of the MSD1 on the short-time behavior of the MSD. For $\Delta t=0.01$ the CBD was still not able to reproduce the true MSD and the GB needs more than $10\Delta t$ to reach the exact curve. Instead the SRK and SE algorithms produce results that follow the correct curve very well. For $\Delta t>0.02$ the third-order corrections start to play an important role and all the MSD1 start to differ. As a consequence, the SE and SRK curves deviate from the exact curve. It should be placed in perspective, however, that these deviations are always more than ten times smaller than those from the CBD scheme at the same size of the time step. Therefore, the SRK and SE algorithms are considerable improvements on the CBD algorithm.

Static properties are determined fully by the static probability distribution. The extent to which the static distribution function calculated by the various algorithms differs from the exact form given by Eq. (3) is shown in Fig. 3. The results are shown for one value of Δt, but the situation for other values is qualitatively very similar. It appears from the

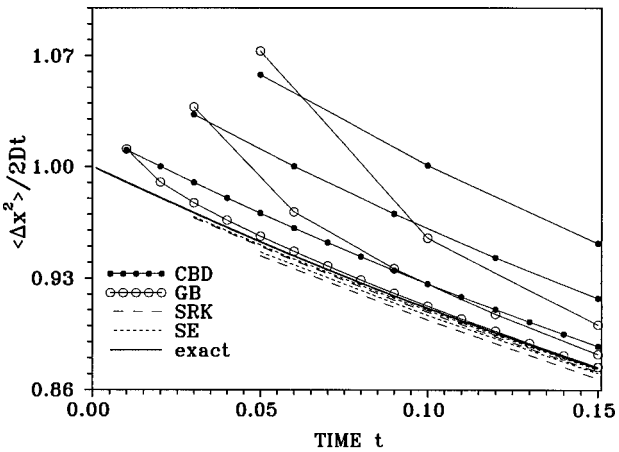

FIG. 2. Short-time region of the MSD from different algorithms and three different sizes of the time step in decreasing magnitude of deviation: 0.05, 0.03, and 0.01. The SE and SRK algorithm data for $\Delta t=0.01$ coincide with the exact curve.

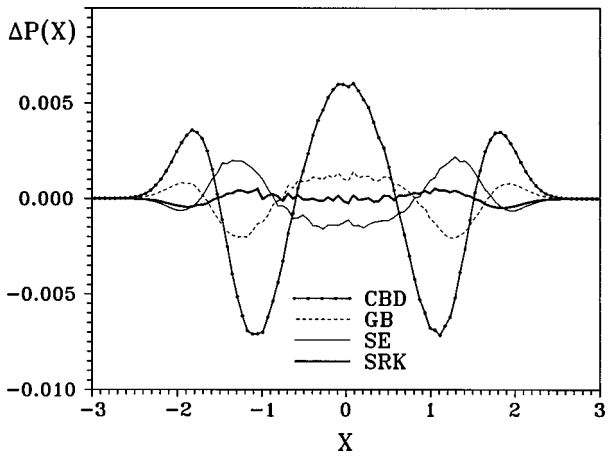

FIG. 3. Deviations of the calculated static probability distribution from its exact form [given by Eq. (3)] for various BD algorithms (the data are for $\Delta t = 0.03$).

figure that the best approximation, at a given value of Δt, is achieved by the SRK method and the worst (as expected) is by the CBD route. The distributions produced by the GB and SE schemes, apart from having the opposite sign, are very similar. This suggests that the accuracy of the static quantities calculated by these methods would also be quite similar. This is, to a certain extent, what we see in Figs. 4(a) and 4(b), where examples for two basic static quantities are presented, the energy and its variance, respectively. For this potential field both properties are related to the fourth and eighth moments of $P(x)$. Notice that, despite a good estimation of $P(x)$, the SRK approach yields a considerably less accurate energy and only a slightly better estimate for the energy variance when compared with the SE method. The advantage of the SE algorithm in estimating the energy follows from mutual cancellations of the $P(x)$ oscillations in evaluating the lower moments. For higher moments the cancellation is not so good and the best estimation comes now (marginally) from the SRK method [see Fig. 4(b)].

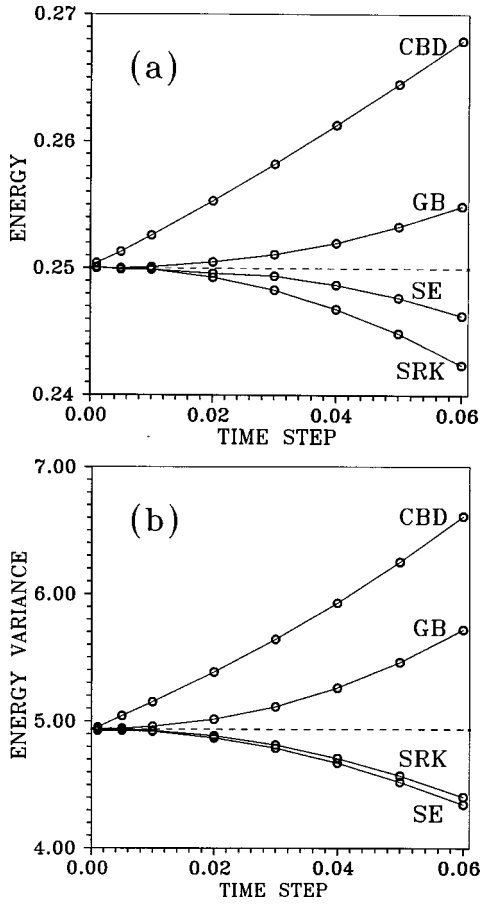

FIG. 4. (a) Energy and (b) its variance of the biharmonic Langevin oscillator vs the time step from the different algorithms.

IV. CONCLUSIONS

In the work we have considered algorithms for solving the stochastic differential equation of the position Langevin equation in its one-variable version. In particular, properties of the biharmonic Langevin oscillator were analyzed in detail from the point of view of algorithmic efficiency at different time steps.

As expected, the original 1975 first-order Brownian dynamics algorithm (CBD) due to Ermak is fairly stable but yields the worst estimate of the calculated quantities for any Δt. It generates with increasing Δt a significant overestimation of both static and dynamic quantities. The GB algorithm gives the largest deviation for the mean square displacement in a time step (MSD1) and, as for the CBD method, is not able to reproduce correctly the short-time region of the mean square displacement. Thus both of these algorithms should be used with caution when accurate results for dynamical quantities at short and intermediate times are required. The GB algorithm, however, yields a much better estimation of static quantities than the CBD algorithm. Such behavior follows probably from the fact that although the GB algorithm is higher order than the CBD algorithm it is not a true second-order algorithm as the deterministic part is of order Δt^2 and the stochastic part involves only a term at the $\Delta t^{1/2}$ level.

The two other position-update schemes considered, the stochastic Runge-Kutta algorithm and the stochastic equation based on Eq. (8) in the text, instead are true second-order algorithms. Both give the correct form for the MSD1 and the best estimation for the mean square displacements. The SE approach gives also the best estimation for the energy and is comparable to the SRK estimate for the energy variance. The form of the error of the MSD1 and its systematic influence on the calculated quantities implies that efficiency of all studied algorithms decreases as the interaction potential becomes less soft. The basis of the present paper relies on the colloid-colloid part of the potential being differentiable. (Algorithms for treating discontinuous potentials are less well developed and require quite different procedures [15,16]. There are serious problems with hard-core discontinuous potentials when short-range fluid lubrication hydrodynamics are included [17,18].)

The SE algorithm gave similar accuracy to that of the CBD algorithm with a time step some 5–10 times larger. Despite some extra computational requirements needed to calculate force derivatives and correlated random number, the resulting performance of the SE is more than three times more efficient than the CBD algorithm. Data from this study and preliminary results on many variable systems (e.g., 120

particles in two dimensions) indicate that the SRK and SE algorithms would appear to have more or less the same computational requirements per step.

The differences between the SE and SRK algorithms come from the fact that the SRK algorithm is an approximation of Eq. (8) in which all of the random terms are represented by a single random number term. Our results suggest that a more rigorous implementation of this expansion in general gives improvements in the accuracy of the calculated quantities. It should be also noticed that the SE method requires only one evaluation of the force loop per time step, which is an advantage when compared to the SRK approach, particularly for larger many-particle systems. A disadvantage of the SE method is that higher-order derivatives of the interaction potential are involved in the calculation and evaluation of correlated random numbers is necessary. Also an extension of the SE approach to deal with position-dependent diffusion coefficients seems to be rather difficult as the stochastic part of the expansion becomes prohibitively more complicated. Work on a many-particle version of the SE scheme is in progress and the results of preliminary calculations support most of the conclusions obtained for the one-variable case described here.

ACKNOWLEDGMENTS

The work was supported by the Polish Committee for Scientific Research (KBN) under Grant No. 8T11F 010 08p04. Some of the calculations were performed at the Poznań Computer and Networking Center.

[1] D. L. Ermak, J. Chem. Phys. **62**, 4189 (1975).
[2] G. C. Ansell and E. Dickinson, J. Chem. Phys. **85**, 4079 (1986).
[3] F. G. Diaz and J. Garcia de la Torre, Macromolecules **23**, 3144 (1990).
[4] J. R. Melrose and D. M. Heyes, J. Colloid Interface Sci. **157**, 227 (1993).
[5] W. Yu and Ch. F. Wong, J. Phys. Chem. **100**, 15 280 (1996).
[6] E. Helfand, J. Chem. Phys. **69**, 1010 (1978).
[7] A. Iniesta and J. Garcia de la Torre, J. Chem. Phys. **92**, 2015 (1990).
[8] R. L. Honeycutt, Phys. Rev. A **45**, 600 (1992).
[9] G. Chirico and J. Langowski, Macromolecules **25**, 769 (1992).
[10] Th. Kirchhoff, H. Löwen, and R. Klein, Phys. Rev. E **53**, 5011 (1996).
[11] W. F. van Gunsteren and H. J. C. Berendsen, Mol. Phys. **45**, 637 (1982).
[12] X. Y. Chang and K. F. Freed, Chem. Eng. Sci. **49**, 2821 (1994).
[13] C. W. Gardiner, *Handbook of Stochastic Methods for Physics, Chemistry and the Natural Sciences* (Springer-Verlag, Berlin, 1985).
[14] R. J. A. Tough, P. N. Pusey, H. N. W. Lekkerkerker, and C. van den Broeck, Mol. Phys. **59**, 595 (1986).
[15] B. Cichocki and K. Hinsen, Physica A **166**, 473 (1990).
[16] D. M. Heyes and J. R. Melrose, J. Non-Newtonian Fluid Mech. **46**, 1 (1993).
[17] D. I. Dratler and W. R. Schowalter, J. Fluid Mech. **325**, 53 (1996).
[18] R. C. Ball and J. R. Melrose, Physica A **247**, 444 (1997).