Algorithms for Brownian dynamics computer simulations: Multivariable case

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 16 February 1999)

Several Brownian numerical schemes for treating stochastic differential equations at the position Langevin level are analyzed from the point of view of their algorithmic efficiency for large-N systems. The algorithms are tested using model colloidal fluids of particles interacting via the Yukawa potential. Limitations in the conventional Brownian dynamics algorithm are shown and it is demonstrated that much better accuracy for dynamical and static quantities can be achieved with an algorithm based on the stochastic expansion and second-order stochastic Runge-Kutta algorithms. The importance of the various terms in the stochastic expansion is analyzed, and the relative merits of second-order algorithms are discussed. [S1063-651X(99)03108-6]

PACS number(s): 02.70.−c

I. INTRODUCTION

Dispersed systems such as polymer solutions and colloidal liquids can be represented by a set of stochastic equations in which the effects of the large number of solvent molecules on polymer or colloidal particles are represented by random forces and frictional terms. The complexity of such systems prohibits exact analytic treatments in all but the most idealized of cases (e.g., infinitely dilute systems). As a result, various problems in dispersed phase systems require computer simulations to solve them. Compared to the well-established techniques for solving deterministic equations of motion, the methods for solving stochastic equations, which are often called stochastic dynamics (SD), are considerably less well developed.

The most simple form of SD called the Brownian dynamics (BD), which has been the mainstay of colloid modeling over the past two to three decades, is the low-order algorithm invented by Ermak and McCammon [1,2]. This technique is at the level of the first-order Euler method for ordinary differential equations and requires a very small time step to produce sufficiently accurate results. The method benefits from its simplicity and is straightforward to use but, because of the small time step required, is quite inefficient. BD simulations that are up to two orders of magnitude longer than those for the equivalent MD systems are required. The BD method can be made more efficient by adopting an appropriate second- or higher-order algorithm. Unfortunately, only a few proposals have been made, with little concern about their ability to handle efficiently large physically relevant many-body systems, and as shown below none of them can be considered as completely satisfactory.

Many of the improved BD schemes are Runge-Kutta-like algorithms with some stochastic terms. Several algorithms along these lines have been proposed, e.g., by Helfand [3], Iniesta and Torre [4], and recently one for the one-variable case by Honeycutt [5]. All these methods, as for the deterministic Runge-Kutta methods, require more than one evaluation of the particle force per time step, which clearly reduces their efficiency.

An algorithm that needs only one evaluation of the force per time step was proposed by van Gunsteren and Berendsen [6]. In the limit of the large friction (i.e., the limit for BD applications), the algorithm has a particularly simple form, which has been used to model, for example, polymer dynamics in solution [7].

In the present work we shall consider, from the point of view of the BD algorithms, a basic finite step-size expansion for the stochastic differential equations, and make a comparison between the efficiency of the different BD algorithms for large or multivariable systems. In a previous publication we carried out a preliminary study for one-dimensional systems [8].

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

II. THE ALGORITHMS

The dynamics of N interacting colloidal particles for many purposes is adequately described by the position Langevin equation,

$$\frac{dr_{i\alpha}}{dt}=\frac{D}{k_BT}F_{i\alpha}+D^{1/2}\xi_{i\alpha}, \tag{1}$$

where $i=1,\ldots,N$ labels the particles and α,β,γ refer to the Cartesian coordinates. The quantity $\xi(t)$ represents a Gaussian white noise process, D is the free-particle self-diffusion constant, k_B is Boltzmann's constant, and T is the temperature. $F_{i\alpha}$ is the net force acting in direction α on the particle, i, derived from the interparticle potential $U(\mathbf{r}^N)$ usually represented as a sum of pairwise additive direct interactions $V(r)$ between the particles,

$$\mathbf{F}_i=-\nabla_iU=-\nabla_i\sum_{j\neq i}^{N}V(|\mathbf{r}_i-\mathbf{r}_j|). \tag{2}$$

The set of equations (1) constitutes a stochastic description of the N-particle evolution through time and space, which is

 © 1999 The American Physical Society

equivalent to the Smoluchowski equation without hydrodynamic interactions. The stationary solution of the Smoluchowski equation is the canonical ensemble distribution,

$$P(\mathbf{r}^N,\infty)=\frac{1}{Z}e^{-U/k_BT}, \tag{3}$$

where Z is the partition function and thus the time averages produced by Eq. (1) are the canonical ensemble averages. From the Smoluchowski or the Langevin equations, the short-time behavior of time correlation functions can be estimated [9]. Explicit results for the form of the Cartesian components of the mean square displacement are available to fourth order in time,

$$\langle \Delta r_{i\alpha}^2\rangle = 2Dt-\frac{D^2}{T}\left\langle\frac{\partial^2 U}{\partial r_{i\alpha}^2}\right\rangle t^2 + \frac{D^3}{3T^2}\sum_{j=1}^{N}\sum_{\beta=1}^{d}\left\langle\left(\frac{\partial^2 U}{\partial r_{i\alpha}\partial r_{j\beta}}\right)^2\right\rangle t^3 - Bt^4+O(t^5), \tag{4}$$

where d denotes the dimensionality of the system, T here (and subsequently) denotes k_BT, and B contains higher derivatives of the total potential energy (the explicit form of this term is irrelevant here).

The conventional Brownian dynamics (CBD) algorithm solves Eq. (1) for the many-body system according to the following particle update scheme:

$$r_{i\alpha}(\Delta t)=r_{i\alpha}+\frac{D}{T}F_{i\alpha}\Delta t+W_{i\alpha}, \tag{5}$$

where Δt is the time step, $r_{i\alpha}\equiv r_{i\alpha}(0)$, and W_i is a normally distributed random force with zero mean and $\langle W_{i\alpha}W_{j\beta}\rangle = 2D\Delta t\,\delta_{ij}\delta_{\alpha\beta}$. van Gunsteren and Berendsen (GB) proposed the following algorithm:

$$r_{i\alpha}(\Delta t)=r_{i\alpha}+\frac{D}{2T}(2F_{i\alpha}+\Delta t\dot{F}_{i\alpha})\Delta t+W_{i\alpha}, \tag{6}$$

where the time derivative of the force is conventionally approximated by $\dot{\mathbf{F}}=[\mathbf{F}(t)-\mathbf{F}(t-\Delta t)]/\Delta t$.

The second-order stochastic Runge-Kutta (SRK) algorithm updates particle positions

$$r_{i\alpha}(\Delta t)=r_{i\alpha}+\frac{D}{2T}(F_{i\alpha}^a+F_{i\alpha}^b)\Delta t+W_{i\alpha}, \tag{7}$$

calculating the forces in two stages $\mathbf{F}_i^a=\mathbf{F}_i(\mathbf{r}^N)$, and then $\mathbf{F}_i^b=\mathbf{F}_i(\mathbf{R}^N)$, at $\mathbf{R}_i=\mathbf{r}_i+(D/T)\mathbf{F}_i\Delta t+\mathbf{W}_i$. For a general stochastic differential equation such as Eq. (1), the following expansion for the Cartesian components of the particle position holds [10]:

$$r_{k\alpha}(\Delta t)=r_{k\alpha}+W_{k\alpha}+\frac{D}{T}F_{k\alpha}\Delta t+\frac{D}{T}\sum_{j=1}^{N}\sum_{\beta=1}^{d}\frac{\partial F_{k\alpha}}{\partial r_{j\beta}}K_{j\beta} + \frac{D^2}{2T^2}\Delta t^2\sum_{j=1}^{N}\sum_{\beta=1}^{d}\frac{\partial F_{k\alpha}}{\partial r_{j\beta}}F_{j\beta} + \frac{D}{2T}\sum_{p,j=1}^{N}\sum_{\beta,\gamma=1}^{d}\frac{\partial^2 F_{k\alpha}}{\partial r_{p\gamma}\partial r_{j\beta}}G_{pj}^{\beta\gamma}, \tag{8}$$

where K and G are random numbers involving W,

$$K_{i\alpha}=\int_0^{\Delta t}W_{i\alpha}(s)ds, \qquad G_{ij}^{\alpha\beta}=\int_0^{\Delta t}W_{i\alpha}(s)W_{j\beta}(s)ds. \tag{9}$$

The expansion of Eq. (8), which we call the SE expansion, results from the integration of Eq. (1), the Taylor expansion for F, and its repeated insertion into itself [10]. It is important to recognize that the random number $K_{i\alpha}$ is also, like $W_{i\alpha}$, a Gaussian random number with the following properties:

$$\langle K_{i\alpha}\rangle=0, \quad \langle K_{i\alpha}K_{j\beta}\rangle=\frac{2}{3}D\Delta t^3\delta_{ij}\delta_{\alpha\beta}, \qquad \langle W_{i\alpha}K_{j\beta}\rangle=D\Delta t^2\delta_{ij}\delta_{\alpha\beta}. \tag{10}$$

Thus, the term in Eq. (8) involving K 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 and the exact formula is replaced by simpler (local) expressions with the same first moment, $G_{ij}^{\alpha\beta}\approx \frac{1}{2}\Delta t W_{i\alpha}W_{j\beta}$.

In the following, the algorithm based on the expansion of Eq. (8) will be called the stochastic expansion (SE) algorithm. In order to see the significance of the various terms in the SE expansion, we shall also consider another algorithm, SE*b*, with the order $\Delta t^{3/2}$ in which the two terms of the second-order have been neglected, and the SE*c* algorithm, in which only one term, the stochastic second-order term, was omitted (the first three terms are just the CBD update scheme). The SE [8] and SE*b* [11] algorithms have already been considered for one-dimensional systems showing several advantages over other schemes and suggesting their potential utility for multivariable systems.

The formal difference between the SRK and SE approaches can be deduced by expanding the $\mathbf{F}^b$ force in the SRK scheme [10]. The resulting formula is very similar to that in Eq. (8) apart from the $\Delta t^{3/2}$ terms in which $K_{i\alpha}$ are approximated by $\frac{1}{2}\Delta t W_{i\alpha}$. It can be shown that, as long as the expansion parameter is small, the following SRK-like scheme gives exactly the SE expansion (8),

$$r_{i\alpha}(\Delta t)=r_{i\alpha}+\frac{D}{2T}(F_{i\alpha}^a+F_{i\alpha}^b)\Delta t+W_{i\alpha} + \frac{D}{2T}\sum_{j=1}^{N}\sum_{\beta=1}^{d}\frac{\partial F_{i\alpha}^a}{\partial r_{j\beta}}S_{j\beta}, \tag{11}$$

where

$$\langle S_{i\alpha}\rangle = 0, \quad \langle S_{i\alpha}S_{j\beta}\rangle = \frac{2}{3}D\Delta t^3\delta_{ij}\delta_{\alpha\beta}, \quad \langle W_{i\alpha}S_{j\beta}\rangle = 0. \tag{12}$$

It is to be noted that $S_{i\alpha}$, like $K_{i\alpha}$, is a Gaussian random number but, unlike $K_{i\alpha}$, is not correlated with $W_{i\alpha}$. The algorithm based on Eq. (11) we shall call the SRK*b* algorithm.

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

$$\langle \Delta r_{i\alpha}^2\rangle = 2D\Delta t - \frac{D^2}{T}\left\langle \frac{\partial^2 U}{\partial r_{i\alpha}^2}\right\rangle \Delta t^2 + \delta_{\mathrm{CBD}}\Delta t^2, \tag{13}$$

and the error in the second-order term is

$$\delta_{\mathrm{CBD}} = \frac{D^2}{T}\left[\frac{1}{T}\left\langle \left(\frac{\partial U}{\partial r_{i\alpha}}\right)^2\right\rangle + \left\langle \frac{\partial^2 U}{\partial r_{i\alpha}^2}\right\rangle\right]. \tag{14}$$

For many physical realizations, the following relation holds:

$$\langle F_{i\alpha}^2\rangle - T\left\langle \frac{\partial^2 U}{\partial r_{i\alpha}^2}\right\rangle = 0, \tag{15}$$

which implies $\delta_{\mathrm{CBD}} > 0$. Thus, the CBD algorithm *always* overestimates the MSD1 by $2D^2\Delta t^2\langle F_{i\alpha}^2\rangle/T^2$.

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

$$\langle \Delta r_{i\alpha}^2\rangle = 2D\Delta t - \frac{D^2}{T}\left\langle \frac{\partial^2 U}{\partial r_{i\alpha}^2}\right\rangle \Delta t^2 + \delta_{\mathrm{GB}}\Delta t^2, \tag{16}$$

where

$$\delta_{\mathrm{GB}} = \delta_{\mathrm{CBD}} + \frac{3D^2}{2T^2}\langle F_{i\alpha}^2\rangle\left(1 - \frac{\langle F_{i\alpha}(\Delta t)F_{i\alpha}(0)\rangle}{\langle F_{i\alpha}^2\rangle}\right). \tag{17}$$

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

$$\langle \Delta r_{i\alpha}^2\rangle = 2D\Delta t - \frac{D^2}{T}\left\langle \frac{\partial^2 U}{\partial r_{i\alpha}^2}\right\rangle \Delta t^2 - \delta_{\mathrm{SRK}}\Delta t^2, \tag{18}$$

with an error contribution,

$$\delta_{\mathrm{SRK}} = \frac{D^3}{2T^3}(\langle F_{i\alpha}^2\rangle - \langle F_{i\alpha}F_{i\alpha}^b\rangle), \tag{19}$$

which is always positive and for small Δt can be approximated by the linear term. Thus, the SRK algorithm is the first algorithm to give the correct second-order term for MSD1 and yields an underestimation of MSD1 with a leading term of order Δt^3. The same is true also for the SRK*b* algorithm.

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

$$\langle \Delta r_{i\alpha}^2\rangle = 2D\Delta t - \frac{D^2}{T}\left\langle \frac{\partial^2 U}{\partial r_{i\alpha}^2}\right\rangle \Delta t^2 + \frac{D^2}{T}\left[\frac{1}{T}\left\langle \left(\frac{\partial U}{\partial r_{i\alpha}}\right)^2\right\rangle - \left\langle \frac{\partial^2 U}{\partial r_{i\alpha}^2}\right\rangle\right]\Delta t^2 + \delta_{\mathrm{SE}}\Delta t^3. \tag{20}$$

On the basis of the relation in Eq. (15), the second term of order Δt^2 is equal to zero (for any Δt) and the expansion of Eq. (8), like the SRK algorithm, gives the correct MSD1 with some deviation only in terms of order Δt^3.

It should be noticed that the same is true also for the SE*b* and SE*c* algorithms.

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 $N = 121$ Brownian particles in two dimensions (2D) interacting via a Yukawa pairwise-additive potential,

$$V(r) = \frac{V_0}{r}\exp[-\lambda(r-1)], \tag{21}$$

where V_0 sets the energy scale and λ is the screening parameter characterizing the steepness and range of the potential. The Yukawa potential, being the electrostatic part of the Derjaguin-Landau-Verwey-Overbeek (DLVO) potential [12], is considered to give a reasonable description of the interaction of a dilute charge-stabilized spherical colloidal suspensions [13], and is often used as a model interaction in BD investigations [14,15].

The basic simulation cell was a square with area A, and the usual periodic boundary conditions were applied. In order to make the system test the various algorithms under demanding conditions, the simulations were performed in the dense fluid region ($T = 1, \varrho = N/A = 0.5$) and the interaction potential was chosen to be strongly repulsive, with $\lambda = 8$.

All quantities presented here are normalized into dimensionless units, by choosing σ, σ^2/D, and V_0/σ as the characteristic values for length, time, and force. In the calculations the averages were calculated from simulations for about 10^3 reduced time periods (i.e., $\sim 10^7$ time steps with a time step $\Delta t = 0.0001$). In the SE algorithm based on Eq. (8), the two correlated random numbers $W_{i\alpha}$ and $K_{i\alpha}$ were sampled from a bivariate Gaussian distribution.

A validation problem with BD is that, unlike MD, there is no conserved quantity that can be used to check the correctness of the time stepping algorithm. Furthermore, in the stochastic part of the BD algorithms a random number generator is used and operations on the random numbers have to be performed. In this situation some subtle error connected with, for example, spurious random number correlations can easily be overlooked and any cross check of the code is highly desirable. In fact, Eq. (15) can serve this purpose. In the case of the SRK algorithm, the equality Eq. (15) is well

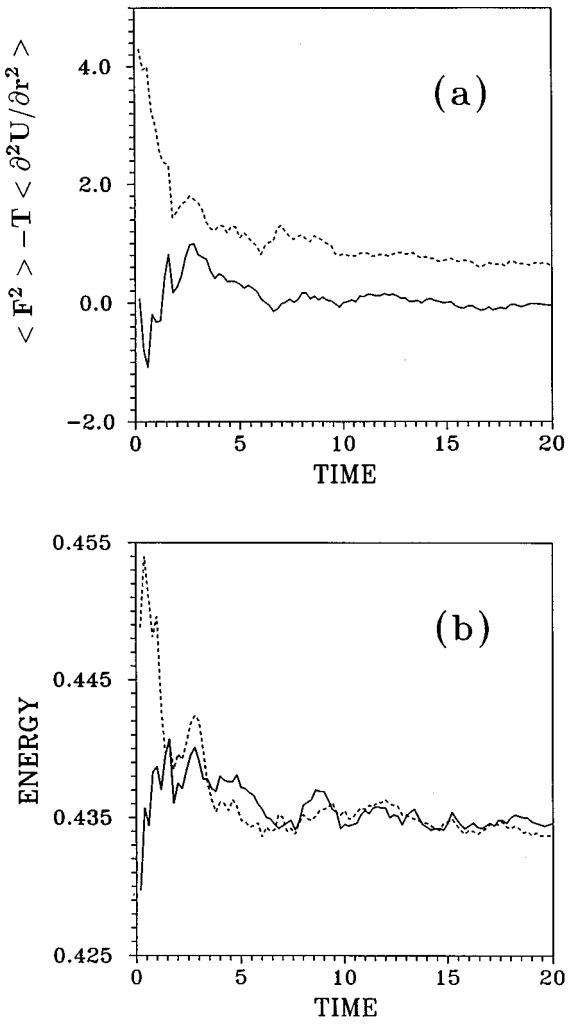

FIG. 1. Test of the cross-check formula Eq. (15). The data are obtained from the SRK scheme with $\Delta t = 2 \times 10^{-5}$. The continuous line represents the data obtained by the correct SRK code and the dashed line is the data produced by the SRK code in which incorrectly the same random number was used in the calculation of the x and y components of the particle displacement. The formula of Eq. (15) vs the accumulated simulation time (a) and the corresponding total energy per particle (b).

obeyed with accuracy better than 0.5% for all $\Delta t \leqslant 0.0005$. An example of utility of Eq. (15) is shown in Fig. 1. In fact, using this relation we were able to detect an error in the SRK code connected with the use of the same random number for x and y coordinates, and which is hardly visible in the other more usually calculated quantities, e.g., energy in Fig. 1(b).

Perhaps the most important yet simplest dynamical quantity is the one we shall consider, and that is the time-dependent mean-square displacement (MSD), over time t, for an arbitrary particle,

$$D(t) = \frac{1}{4t} \langle [\mathbf{r}(t) - \mathbf{r}(0)]^2 \rangle, \tag{22}$$

which is averaged over all particles.

In the long-time limit it gives the self-diffusion coefficient of the particle. The general behavior of the MSD calculated with the various algorithms is shown in Fig. 2 for a fairly large time step $\Delta t = 0.0005$ (for other time steps the results

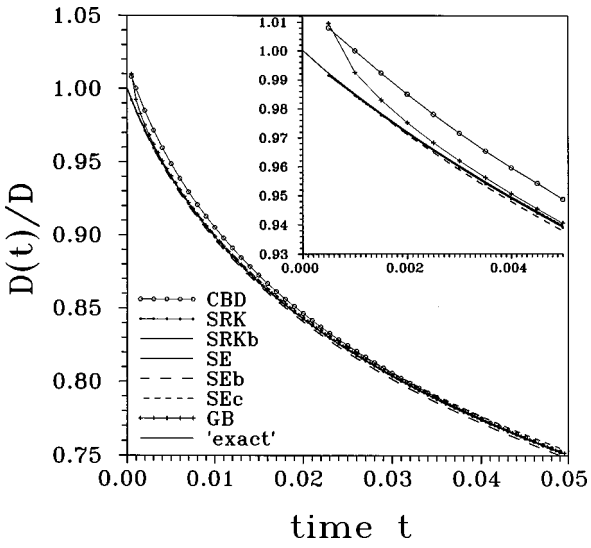

FIG. 2. The mean square displacement over time of a 2D Yukawa fluid obtained from the seven BD algorithms discussed in the text with time step $\Delta t = 0.0005$. The curve labeled ''exact'' is the limit obtained from calculations with $\Delta t = 2 \times 10^{-5}$. The SE, SRK, and SRK*b* algorithm data coincide with the exact curve. The inset presents an enlargement of the short-time regions.

are qualitatively very similar). On the scale of the figure the MSD curves obtained from the SRK, SRK*b*, and SE calculations coincide with the exact curve, produced by all algorithms in the limit of very small time step. The CBD curve very slowly approaches from above towards 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 Δt) to the correct form. Notice that in accordance with Eq. (17), it starts above the CBD curve. The SE*b* produces the correct short time limit in accordance with Eq. (20) but fairly quickly starts to bend downward with the maximum deviation somewhere around $t = 0.05$ and the deviations from the exact curve seem to persist for a long time. Also the SE*c* scheme gives noticeable deviations from the exact MSD curve. As one may see in Fig. 2, apart from the short-time region, it consequently overestimates the exact results.

The statistical uncertainty of the MSD data, as for any autocorrelation function (ACF), increases with time from the origin. In our case, in the long-time region ($t > 2$) where a plateau is reached, all calculated MSD curves (at least for $\Delta t \leqslant 0.0005$) lie within the error bars around the ''exact'' curve. Only the data produced by the CBD algorithm with $\Delta t = 0.0005$ seem to show some systematic departure and deviations at long times. In this situation it is difficult to make definitive statements about the influence of the particular algorithm on the long-time behavior of the MSD.

Therefore, within statistical uncertainty, all algorithms reproduce the long-time behavior of the MSD, but differ considerably in their ability to reproduce its intermediate and particularly short-time characteristics. This is clearly seen in the enlargement in Fig. 2, which illustrates a significant influence of the MSD1 on the short-time behavior of the MSD. In the enlargement, slight differences between the SRK, SRK*b*, and SE algorithms are still hard to notice. The SE results practically coincide and those from the SRK method

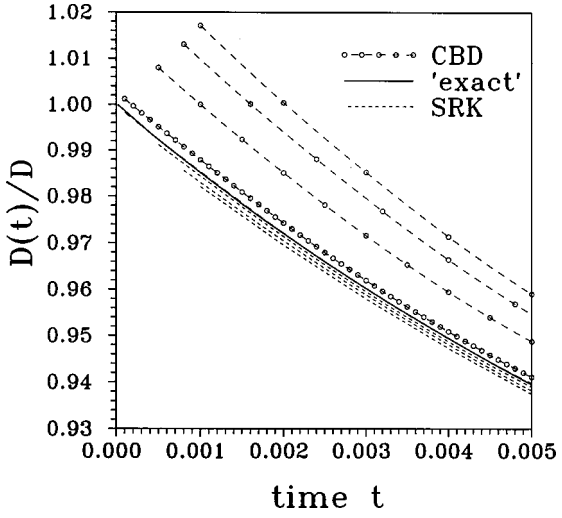

FIG. 3. The short-time region of the *msd* from the SRK and CBD algorithm and four different sizes of the time step in increasing magnitude of deviation: 0.0001, 0.0005, 0.0008, and 0.001.

only very slightly underestimate the exact curve. The SRK*b* method produces data which lie between the SE and SRK curves.

The significant influence of the size of the time step on the MSD1 and the short-time behavior of the MSD is illustrated for the SRK and CBD algorithms in Fig. 3. The figure also demonstrates that on increasing the magnitude of Δt, deviations of the SRK data from the exact curve emerge, although the SRK deviations are always more than ten times smaller than those from the CBD algorithm at the same size of the time step.

The second important dynamical quantity of primary interest often calculated in BD simulations is the shear-stress time autocorrelation function (SACF), defined as

$$C(t)=\frac{V}{\varrho T}\langle\sigma_{\alpha\beta}(0)\sigma_{\alpha\beta}(t)\rangle, \tag{23}$$

where $\sigma_{\alpha\beta}$ is an off-diagonal component of the stress tensor ($\alpha\neq\beta$). The $C(t)$ determines, through the Green-Kubo relation, the shear viscosity and its initial value gives the infinite-frequency shear modulus $G_\infty=C(0)$. The shear stress correlation function is a collective quantity, which decays relatively quickly towards zero. The normalized SACFs calculated with the various algorithms are compared in Fig. 4. At short times the $C(t)$ produced by the different algorithms converge to the exact curve, which is quite different behavior to what was observed for the MSD. With increasing time the curves become more different ($t\sim 0.03$) and at even longer times ($t>0.1$) they converge slowly again towards the exact curve where they are mutually consistent within error bars. The most significant deviations are therefore at intermediate times, which may be seen more clearly in the inset in Fig. 4. Similarly, as in the case of the MSD, the largest deviations are produced by the CBD, SE*b*, and SE*c* algorithms, although now the CBD algorithm underestimates and SE*b* overestimates the exact curve. Also the GB scheme produces noticeable deviations from the exact curve. Again the best estimate is given by the SRK, SE, and SRK*b* algorithms.

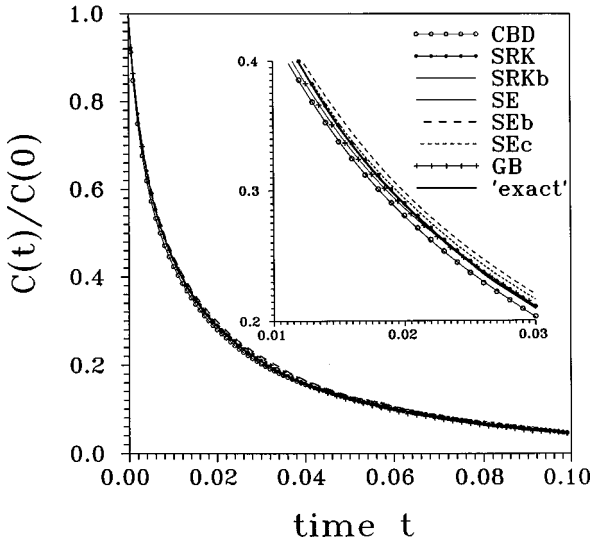

FIG. 4. The SACF from different algorithms and $\Delta t=0.0005$. The SE, SRK, and SRK*b* data coincide with the ''exact'' curve and the inset shows an enlargement of the intermediate-time regions.

The influence of different sizes of the time step on the SACF is illustrated in Fig. 5 for the case of the CBD and SRK algorithms. In the case of the SRK scheme, apart from results obtained with the largest time steps $\Delta t>0.0005$, all the data practically coincide with the exact curve (similar behavior is observed for the SE and SRK*b* schemes). In contrast, the influence of the magnitude of Δt on the CBD scheme is quite significant and, as may be seen from the figure, only calculations with $\Delta t\leqslant 0.0001$ lead to the correct form of the SACF (and consequently the viscosity of the system).

The behavior of the static quantities obtained by different algorithms at various time steps is illustrated in Figs. 6 and 7. In Fig. 6 the total energy per particle is shown. It appears from the figure that the best approximation, at a given size of the time step, is achieved by the SE method and the worst (as

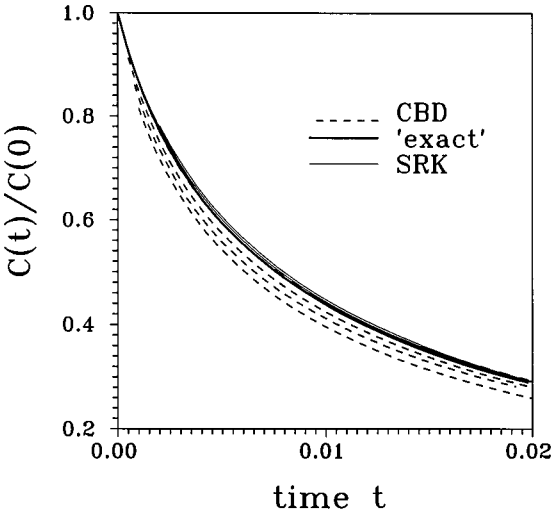

FIG. 5. The SACF from the SRK and CBD algorithm. The results are for different sizes of the time step in increasing magnitude of deviation: 0.0001, 0.0005, 0.0008, 0.001. For $\Delta t\leqslant 0.0005$ the SRK data coincide with the ''exact'' curve.

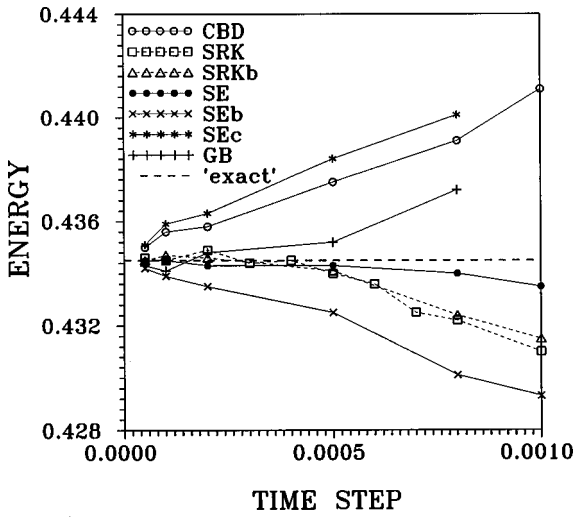

FIG. 6. Energy of the Yukawa fluid vs time step from the different algorithms. The exact value is estimated to be 0.4345.

expected) is by the CBD route. For $\Delta t \leqslant 0.0005$, the results produced by the SRK and SRK*b* algorithms are practically as good as the SE (within the error bars they all coincide with the exact values). For the larger time steps the deviations from the exact value become more significant. The results produced by the SE*b* method are only slightly better and by the SE*c* even slightly worse than that obtained by the CBD route. The SE*c* method becomes unstable at $\Delta t > 0.0008$. All this clearly demonstrates the great importance of the particular terms in the SE expansion. The GB approach gives a fairly good estimation of the energy, although the method becomes unstable for $\Delta t > 0.0008$. The results for the pressure are qualitatively very similar to that for the energy. As one can see in Fig. 7, the results for the shear modulus are quite similar to those of the energy. Again, the worst estimates come from the CBD, SE*c*, and SE*b*, and the best from SE, SRK, and SRK*b* schemes.

The clear disability of the SEc approach to estimate properly the static quantities is rather unexpected. To confirm the important role of the second-order non-Gaussian stochastic term in the SE expansion, we have also performed BD calculations with a 1D Brownian particle in a biharmonic potential i.e., the system studied by us recently [8]. We confirmed that as for the multivariable case, also in the one-variable system, the energy produced by the SE*c* algorithm with different values of Δt is very similar to that produced by the CBD method (i.e., considerably worse than that obtained by the SE, SEK, and GB methods). Thus, the second-order stochastic term in the SE expansion seems to be important irrespective of the complexity of the system studied.

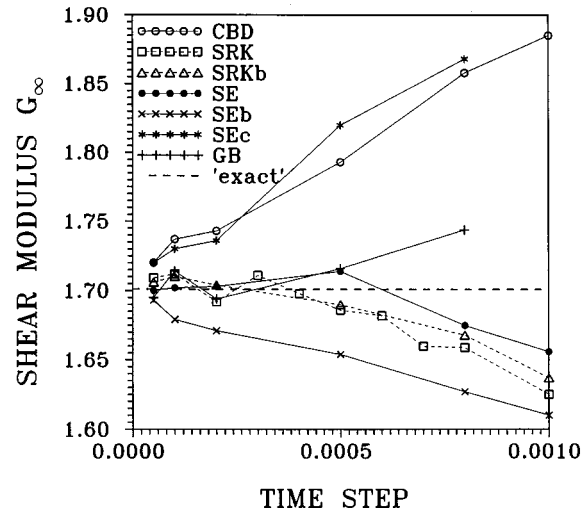

FIG. 7. The shear modulus vs time step from the different algorithms. The exact value is estimated to be 1.70.

It is also apparent, particularly in Figs. 6 and 7, that the second-order schemes SRK, SRK*b*, and SE provide useful information on the maximum value of Δt at which the algorithm can be properly used, enabling us to choose the optimum value of the time step. In the case studied here this maximum time step size is about 0.0005, which is almost an order of magnitude greater than what should be used in the CBD calculations, but it is worth noting that it is still an order of magnitude smaller than can be used in deterministic algorithms for the same potential and thermodynamic parameters.

Finally, we have also performed some calculations with the so-called *predictor-corrector* algorithm proposed by Mannella and Palleschi (MP) [11]. The MP algorithm, like the SRK scheme, needs two evaluations of the force per time step but involves two stochastic terms and thus can be expected to produce more accurate results. Our calculations for the 121 Brownian particle fluid have shown, however, that the data produced by the MP algorithm, with $\Delta t > 0.0002$, are noticeably less accurate than those obtained by the SRK approach, similar to the result we obtained for the 1D Brownian particle system. Thus, in the cases studied by us, the MP was less accurate and thus less efficient than the SRK approach.

IV. CONCLUSIONS

In this work we have considered algorithms for solving the stochastic differential equation of the position Langevin equations. In particular, properties of the Yukawa fluid were analyzed from the point of view of algorithmic efficiency at different time steps.

As expected, the original 1975 first-order Brownian dynamics (CBD) algorithm due to Ermak yields the worst estimate of the calculated quantities for any Δt. It generates with increasing Δt a significant overestimation or underestimation of both static and dynamic quantities. It is, however, the simplest scheme to implement and is fairly stable. The van Gunsteren and Berendesen (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 displacements. It also produces noticeable deviations in the intermediate-time region of the shear autocorrelation function. 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 probably follows from the fact that, although the GB algorithm is higher order than the CBD, 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.

Terms of different levels are involved also in the SE*b* algorithm (the deterministic part here is of order Δt but the stochastic one is of order $\Delta t^{3/2}$) and in the SE*c* algorithm (here the deterministic part is of order Δt^2 and the stochastic one is of order $\Delta t^{3/2}$). In contrast to the GB scheme, they give rather unsatisfactory static quantity estimation but good short-time MSD data. Additionally, the noticeable deviations of the MSD and the SACF at intermediate times make the SE*b* and SE*c* approaches rather uncompetitive compared to the second order schemes. The results produced by the SE*b*, SE*c*, and GB as well as by MP schemes indicate that algorithms based on different level terms in the deterministic and stochastic parts can produce less accurate and/or less efficient results than the consistently lower-order algorithms. They seem to be also less stable with increasing Δt. Furthermore, the SE*c* results clearly show a great importance of the non-Gaussian second-order stochastic term in the stochastic expansion of Eq. (8). Any schemes, e.g., higher-order schemes [16], even if used only for a few variable systems, should be applied with caution if this term is neglected.

The second-order position-update schemes considered, the stochastic Runge-Kutta algorithm (SRK), SRK*b*, and the SE based on Eq. (8) in the text, give the correct form for the MSD1 and the best estimation for the mean-square displacements and SACF. The SE approach gives also the best estimation for the energy. The SRK*b* scheme offers a more accurate estimation of calculated quantities than the SRK approach, although the improvement is marginal and in practice it does not compensate for the additional programming effort required to include the extra force terms in the SRK*b* update scheme.

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 rigorous implementation of this expansion gives improvements in the accuracy of the calculated quantities. 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. Our calculations indicate that the large number of rather complicated force-related terms in the SE scheme increases considerably the computational requirements per step. We estimate its computational efficiency to be between the CBD and SRK methods. The SE approach should have advantages over other approaches in the case of few variable problems and/or a very simple form of the interparticle interactions (e.g., the harmonic interactions).

The form of the error of the MSD1 and its systematic influence on the calculated quantities implies that the efficiency of all the studied algorithms decreases as the interaction potential become harder.

For multivariable or larger systems, among the considered approaches, the SRK algorithm seems to be the one which can be recommended. Its main drawback is the double evaluation of the force loop per time step. This roughly doubles the CPU time but, as has been shown above, the SRK scheme produces the data which are several times more accurate than those produced by the CBD scheme and in summary it is at least four times more efficient than the CBD approach.

In conclusion, we have to say that none of the above algorithms can be considered as entirely satisfactory in being accurate, fast, stable, and easy to implement for the BD simulations of multivariable systems. This work clearly indicates that the task of developing an efficient algorithm for the multivariable stochastic differential equations is far from being solved.

ACKNOWLEDGMENTS

The work has been supported by Polish Committee for Scientific Research (KBN) Grant No. 8T11F01214. We are grateful to Professor Tony Ladd, University of Florida at Gainesville, for helpful discussions. Parts of the calculations were performed at the Poznań Computer and Networking Center.

[1] D. L. Ermak, J. Chem. Phys. **62**, 4189 (1975).
[2] D. L. Ermak and J. A. McCammon, J. Chem. Phys. **69**, 1352 (1979).
[3] E. Helfand, J. Chem. Phys. **69**, 1010 (1978).
[4] A. Iniesta and J. Garcia de la Torre, J. Chem. Phys. **92**, 2015 (1990).
[5] R. L. Honeycutt, Phys. Rev. A **45**, 600 (1992).
[6] W. F. van Gunsteren and H. J. C. Berendsen, Mol. Phys. **45**, 637 (1982).
[7] X. Y. Chang and K. F. Freed, Chem. Eng. Sci. **49**, 2821 (1994).
[8] A. C. Brańka and D. M. Heyes, Phys. Rev. E **58**, 2611 (1998).
[9] R. J. A. Tough, P. N. Pusey, H. N. W. Lekkerkerker, and C. van den Broeck, Mol. Phys. **59**, 595 (1986).
[10] A. Greiner, W. Strittmatter, and J. Honerkamp, J. Stat. Phys. **51**, 95 (1988).
[11] R. Mannella and V. Palleschi, Phys. Rev. A **40**, 3381 (1989).
[12] W. Hess and R. Klein, Adv. Phys. **32**, 173 (1983).
[13] H. Löwen, Phys. Rev. Lett. **72**, 424 (1994); J. Chem. Phys. **100**, 6738 (1994).
[14] H. Löwen, J. P. Hansen, and J. N. Roux, Phys. Rev. A **44**, 1169 (1991).
[15] H. Löwen, T. Palberg, and R. Simon, Phys. Rev. Lett. **70**, 1557 (1993).
[16] E. Hershkovitz, J. Chem. Phys. **108**, 9253 (1998).