INTEGRATION OF STOCHASTIC DIFFERENTIAL EQUATIONS ON A COMPUTER

RICCARDO MANNELLA
Dipartimento di Fisica, Università di Pisa, and Istituto Nazionale Fisica della Materia, UdR Pisa
Via Buonarroti 2, 56100 Pisa, ITALY
Email: riccardo.mannella@df.unipi.it

Received (received date)
Revised (revised date)

A brief introduction to the simulation of stochastic differential equations is presented. Algorithms to simulate rare fluctuations, a topic of interest in the light of recent theoretical work on optimal paths are studied. Problems connected to the treatment of the boundaries and correlated noise will be also discussed.

Keywords: Numerical Integration; Stochastic differential Equations; Large fluctuations; Correlated Noise.

1. Introduction

Stochastic processes, introduced to model a variety of different physical situations, are ubiquitous in modern science. Unfortunately, the most common situation is that the "solution" of some quantities connected to the stochastic model cannot be found theoretically: then, one typically turns to simulations, analogue or digital, of the system of interest. A very complete review of analogue techniques has recently appeared, and the interested reader is referred to it for further details [1]. We concentrate here on simulations of stochastic processes on a computer (digital simulations), with particular emphasis on simulations of rare fluctuations. Rare fluctuations are fluctuations which bring the stochastic system very far from the phase space which the system explores most of the time. It is possible to relate the happening of a rare fluctuation to some building up of an activation energy (one can think of the energy necessary to overcome a potential barrier, like in a chemical reaction). The nature of rare fluctuations is such that we should have algorithms which correctly explore the tails of the distribution functions: this is so because the "rare fluctuation" explores the region of low probability. We should be able to stop in a correct way our simulation when the rare fluctuation we are simulating hits a prescribed boundary in phase space. We should optimise, if possible, our algorithms to situations when the system lacks detailed balance, like when the stochastic process driving the system is not white. Finally, we should have pseudo random number generators fast and very reliable, able to provide us with very long random sequences. We will try to address all these problems in this paper. For further comments and references, the interested reader can consult, among others, [2,3,4,5,7].

2. The basic algorithm

A stochastic differential equation has the generic form

$$\dot{x}_i = f_i(\vec{x}) + g_i(\vec{x})\xi(t), \quad \langle \xi(t) \rangle = 0, \quad \langle \xi(t)\xi(s) \rangle = \delta(t-s), \tag{1}$$

where we assume that the stochastic process ξ is Gaussian and that only one stochastic forcing is present. In the following, we call h the integration time step, and we use Stratonovich calculus [4]. A simple minded approach to solve Eq. 1 is to formally integrate it, then to use a Taylor expansion around the point $t = 0$, to find recursively the various contributions [6]. Restricting the discussion to a one dimensional model, the equation has the form

$$\dot{x} = f(x) + g(x)\xi(t) \tag{2}$$

A formal integration yields

$$x(h) - x(0) = \int_0^h \left(f(x(t)) + g(x(t))\xi(t) \right) dt \tag{3}$$

Define

$$f_0 \equiv f(x(0)) \qquad g_0' \equiv \left. \frac{\partial g(x(t))}{\partial x(t)} \right|_{x=x(0)}$$

and so on. By Taylor expansion it is meant that the functions are expanded as $f_t = f_0 + (x(t) - x(0))f_0' + \dots$. The simple minded lowest order in h seems to be

$$x(h) - x(0) = \int_0^h \left(f_0 + g_0\xi(t) \right) dt = hf_0 + g_0 \int_0^h \xi(t)dt. \tag{4}$$

We will see that this is not the correct lowest order in h. For the moment, note that on the r.h.s. there is a so called "stochastic integral"

$$Z_1(h) \equiv \int_0^h \xi(t)dt \tag{5}$$

which is the integral over the time range $(0, h)$ of the stochastic process $\xi(t)$. This integral is a stochastic variable, and the integration amounts at adding up some gaussian variables: as such, $Z_1(h)$ is itself a Gaussian variable, or, in other words, its probability distribution is a Gaussian distribution. This implies that the probability distribution of $Z_1(h)$ is determined once the average and the standard deviation of the distribution are known. A simple minded numerical integrator would then be, at each time step:

- generate a random gaussian variable, with appropriate average and standard deviation (to "simulate" the stochastic integral);
- substitute the stochastic integral on the r.h.s. of Eq. 4 with this random variable;

- integrate the equation using any standard integrator valid for deterministic differential equations.

How can we work out the statistical properties of $Z_1(h)$? We only need its average and its standard deviation. Using $\langle ... \rangle$ to indicate statistical averages,

$$\langle Z_1 \rangle = \int_0^h \langle \xi(s) \rangle ds = 0 \tag{6}$$

$$\langle Z_1^2(h) \rangle = \int_0^h \int_0^h \langle \xi(s)\xi(t) \rangle dsdt = \int_0^h \int_0^h \delta(t-s) dsdt = \int_0^h ds = h. \tag{7}$$

If we introduce a stochastic gaussian variable with average zero and standard deviation one, Y_1, it follows that we can write the following representation for $Z_1(h)$

$$Z_1(h) = \sqrt{h} Y_1$$

meaning that, using this definition, $Z_1(h)$ has the correct statistical properties. We can also rewrite Eq. 4 as

$$x(h) - x(0) = hf_0 + Z_1(h) g_0 = hf_0 + \sqrt{h} g_0 Y_1. \tag{8}$$

A problem is apparent: on the r.h.s., the first term is order of h, but the second one is order of $\sqrt{h}$: in principle, we should insert the x increment from Eq. 4 in Eq. 3, take one more term in the Taylor expansion, and check the order of the contribution we get. It turns out that if we did that, the correct algorithm to first order in h would be

$$x(h) - x(0) = g_0 Z_1(h) + f_0 h + \frac{1}{2} g_0' g_0 Z_1(h)^2 \tag{9}$$

The higher order terms are obtained by recursion, inserting the lower order terms in Eq. 3 and collecting the different contributions. Before we write the algorithm at order of h^2 (which is the highest order we can get recursively), let us introduce other stochastic integrals which are relevant in the derivation. In the following, Y_1 is the same stochastic variable used for $Z_1(h)$, and Y_2 and Y_3 are two more gaussian stochastic variables, with average zero and standard deviation one, independent of each other. We need

$$Z_2(h) = \int_0^h Z_1(s) ds = h^{3/2} \left\{ \frac{Y_1}{2} + \frac{Y_2}{2\sqrt{3}} \right\}$$

$$Z_3(h) = \int_0^h Z_1^2(s) ds \approx \frac{h^2}{3} \left\{ {Y_1}^2 + Y_3 + \frac{1}{2} \right\}$$

For additive noise ($g(x) = \sqrt{2D}$), the h^2 algorithm reads

$$x(h) = x(0) + \sqrt{2D} Z_1(h) + f_0 h + \sqrt{2D} Z_2(h) f_0' + D Z_3(h) f_0'' + \frac{h^2}{2} f_0' f_0 \tag{10}$$

We will call this the "Full algorithm". In the n dimensional case, for one external stochastic forcing which is additive, i.e. for a system described by

$$\dot{x}_i = f_i(\vec{x}) + g_i \xi(t)$$

we find (defining here $f_{i,j} \equiv \partial f_i / \partial x_j$ evaluated in $x_i(t=0)$ etc., and assuming a sum over repeated indeces)

$$\begin{aligned} x_i(h) &= x_i(0) + g_i Z_1(h) + f_i h + Z_2(h) f_{i,k} g_k + \\ &\quad \frac{1}{2} f_{i,jk} g_j g_k Z_3(h) + \frac{1}{2} h^2 f_{i,j} f_j. \end{aligned} \tag{11}$$

Expressions valid for the more general case can be found in [2]. Other integration schemes which can be found in the literature are (see quoted references):

Euler scheme: Eq. 10, keeping only the first three terms on the r.h.s..

"Exact propagator": solve exactly $\dot{x} = f(x)$ and then add $Z_1(h)$ to take into account the noise.

Heun scheme: Use the following integrator:

$$\begin{aligned} x_1 &= x(0) + \sqrt{2D} Z_1(h) + f_0 h \\ x(h) &= x(0) + \sqrt{2D} Z_1(h) + \frac{h}{2} \left(f_0 + f(x_1) \right) \end{aligned}$$

Some authors have developed Runge-Kutta schemes (see [4]). We will see, however, that the particular nature of a SDE is such that higher order schemes may not lead to substantial improvements to the integration.

3. Which is the "best" algorithm?

The problem of the "best algorithm" for a SDE has two aspects: deterministic accuracy and stochastic behaviour. We depart here very much from standard textbooks on stochastic integration. The mathematical literature on stochastic processes ((see, among others, [8]) proceeds on introducing a measure of how closely a given numerical integration scheme approximates the stochastic model one is studing: to this end, the the definitions of weak or strong convergency are introduced. Although we refer the interested reader to any textbook on stochastic process for a formal definition of these different convergence criteria, here we will simply say that these criteria are based on the evaluation, in a statistical sense, of the difference between the "true" evolution and the "simulated" evolution as function of time; this difference (which is a statistical quantity, and it is essentially the norm of the difference between some exact moment and the corresponding evoluted moment) typically increases in time, and its growth is a function of both the integration time step and the noise intensity; the difference between strong and weak convergence is on how the converge to the "true" evolution is evaluated. It must be appreciated that this approach to studying the convergence is only valid for the"short times" dynamics: although this is normally fine, it is obvious that a good behaviour at short times does not imply anything at large times, and in particular it does not imply that the stochastic motion generated by our algorithm faithfully samples the correct equilibrium distribution. Here, as mentioned, a different approach will be followed: the different schemes will be first studied in the absence of noise ("deterministic accuracy"); and subsequently, in the presence of noise, both for short and large times.

3.1. *Deterministic accuracy*

In this case, one disregards the stochastic component, and studies only the deterministic integration scheme, using standard techniques. This gives some indications of the "deterministic accuracy" of the scheme considered. In this case, the usual machinery (used to work out accuracy, stability etc.) applies. The error associated with a given integration scheme is easily evaluated; we find the following:
Euler scheme: accurate up to $O(h)$.
Exact propagator: no numerical error associated with this algorithm (by definition).
Heun scheme: accurate up to $O(h^2)$.
Full algorithm: accurate up to $O(h^2)$. The "Exact propagator" is in practice obtained using a very high order integration scheme, for instance a Runge-Kutta.

3.2. *Stochastic behaviour*

In this case, one can check the Taylor expansion to judge the short times dynamics. It is clear then that the best algorithm in this time range is the "Full algorithm", given that it was derived as a Taylor expansion of the stochastic equations. Alternatively, standard techiques mentioned previously are available: basically, even after introducing converge in a weak or strong sense, we find that the algorithm which has the better converge is the "Full algorithm". Obviously, virtually all of the research in stochastic integration schemes has dealt with the behaviour of the various algorithms in this limit. Some interesting schemes can for instance be found in [9]. This latter reference does not mean to be exaustive, but simply it points to a couple of papers where very many integration schemes are introduced and discussed.

The large times behaviour can be studied, on the other hand, deriving the equilibrium properties from the propagator used in the numerical scheme (there are other possibilities: for instance, one can consider which scheme is the closest to the bona fide trajectory under some measure). We have in mind the problem of rare large fluctuations, so, as a rule of thumb, we should use integrators which reproduce as closely as possible the large time dynamics, i.e. same equilibrium quantities. Focusing on the large times dynamics, the idea is to start from a generic form of the integrator, $\vec{x}(t) = \vec{x}(0) + \vec{F}(\vec{x}, t)$, and, for instance, find the equilibrium distribution it generates; then, compare it to the real one. It follows, writing the propagator, that [10]

$$P(\vec{x}, t+h) - P(\vec{x}, t) = \sum_{n=1}^{\infty} \sum_{x_i} \frac{\partial}{\partial x_1} \dots \frac{\partial}{\partial x_n} K_{1\dots n} P(\vec{x}, t) \tag{12}$$

where $P(\vec{x}, t)$ is the probability distribution generated in the simulations, starting from an initial $P(\vec{x}, 0)$ and

$$K_{1\dots n} \equiv (-1)^n \frac{1}{n!} \langle F_1 \dots F_n \rangle_{noise}.$$

At equilibrium, the difference on the l.h.s. in Eq. 12 is zero, and the r.h.s. of Eq. 12 becomes an implicit equations for $P(x, \infty))$. In general, for systems in detailed balance,

$$P(\vec{x}, \infty)_{sim} = P(\vec{x}, \infty)_{true} \times \exp \sum_{n=1}^{\infty} h^n S_n / D$$

where all S_n would be zero if the algorithm were exact. Focusing on the system

$$\dot{x} = -V'(x) + \sqrt{2D}\xi(t)$$

which has the exact equilibrium distribution

$$P(x, \infty)_{true} = N \exp\{-V(x)/D\}$$

and the actual equilibrium distribution (using the different numerical integration schemes)

$$P(x, \infty)_{sim} = N' \exp\{(-V(x) + hS(h, x))/D\}.$$

Carrying out the necessary algebra, it is straightforward to find the function $S(x)$
Euler scheme: $S(h, x) = (V')^2/4 - DV''/2$
Exact propagator: $S(h, x) = (V')^2/2 - DV''/2$
Heun scheme: $S(h, x) = O(h)$
Full algorithm: $S(h, x) = O(h)$
The results of simulations using $-V'(x) = x - x^3$ are summarised in Fig. 1. It is clear that the "Full algorithm" and the Heun scheme are the algorithms which most faithfully reproduce the equilibrium distribution: note how they nicely reproduce the theoretically expected equilibrium distribution. It is also interesting to note that the "Exact propagator" does not do better than the Euler scheme: this implies that to derive higher order algorithms it is necessary to deal with the higher order terms coming both from the deterministic and the stochastic part of the SDE. But, as we saw in the derivation of the "Full algorithm", at order h^2 we start to have non gaussian stochastic variables ($Z_3(h)$), so higher order schemes may not be well founded. Among others, the higher order algorithms of ref. [9] do not do better (and sometimes they do worse!) than the "Full algorithm" or the "Heun scheme", although they are much more expensive in terms of computing power [11].

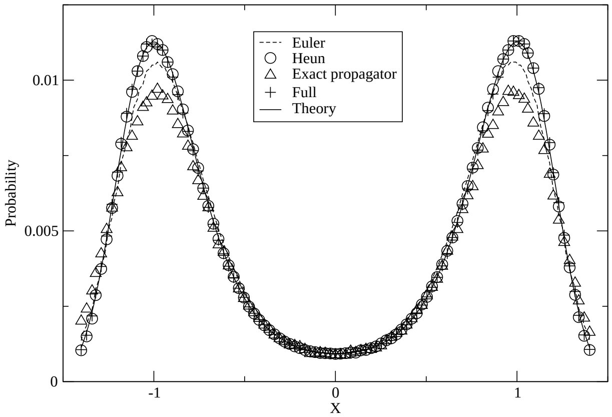

Fig. 1. Comparison between equilibrium distributions obtained using different integration schemes. $D = h = 0.1$

4. The $\sqrt{h}$ problem and boundaries

Regardless of the integration scheme used to carry out the integration, there is an intrinsic problem with a SDE, due to the discreteness of the integration and to the sampling; this is a problem present even with an ideal integrator [13,12]. Fig. 2 shows that a stochastic trajectory appears very different over different time scales (i.e., h's). The problem is most acute when we need to stop the integration because a boundary is reached (see the dashed line in Fig. 2), like, for instance, in a Mean First Passage Time (MFPT) evaluation: the "decimated" trajectory (a trajectory obtained integrating with a larger h) simply misses the transition. The cure is simple: we need to evaluate the probability that the trajectory hit the boundary and came back, within an integration time step, and thus stop the simulations.

Following [12], for a system described by the SDE

$$\dot{x} = F(x) + \sqrt{2D}\xi(t), \quad \langle \xi(t) \rangle = 0, \quad \langle \xi(t)\xi(s) \rangle = \delta(t-s) \tag{13}$$

the probability that the stochastic trajectory, which is at x_0 at time $t = 0$ and at x_h at $t = h$, hit a boundary x_b at an intermediate time is given by

$$\begin{aligned} P(\text{hit}) &= \exp\left[-\frac{F_b'}{2D\left(e^{2hF_b'} - 1\right)} \left(x_h - x_b + (x_0 - x_b)e^{hF_b'} - \frac{F_b}{F_b'} \right)^2 \right. \\ &+ \left. \frac{1}{4Dh} \left(x_1 - \left(x_0 + h\frac{F_0 + F_h}{2} \right) \right)^2 \right] \end{aligned} \tag{14}$$

where $F_b = F(x_b)$ etc.. At each integration time step Eq. 14 is evaluated, and a uniformly distributed random variable is generated in the range (0,1). If the random variable is smaller that Eq 14, it is assumed that the trajectory hit the boundary

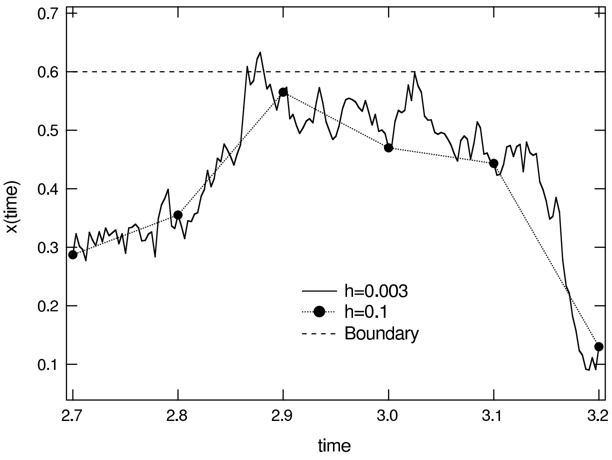

Fig. 2. Comparison between trajectories done with different time steps

and came back, and the appropriate action for reaching the boundary should be taken.

4.1. *Free diffusion with absorbing boundaries*

As a test system, take a bunch of particles injected at the $x = 0$, which can freely move in one dimension, until they reach the boundaries, located at $\pm L$, where they are absorbed [13]. The ruling SDE is

$$\dot{x} = \sqrt{2D}\xi(t), \quad \langle \xi(t) \rangle = 0, \quad \langle \xi(s)\xi(t) \rangle = \delta(t - s). \tag{15}$$

For $h = 0$, the MFPT to the boundary is $\tau(0) = \frac{L^2}{2D}$, whereas, for a finite h, it becomes [13]

$$\tau(h)/\tau(0) = 1 + \sqrt{32/9\pi}\sqrt{h/\tau(0)}. \tag{16}$$

Figure 3 shows that indeed the MFPT's simulated without correction follow Eq. 16. As soon as the correction for the finiteness of the integration time step is introduced, the agreement between simulations and $h = 0$ theoretical MFPT becomes excellent (lower circles). Let us stress that the integration of this dynamical system is exact, due to the structure of Eq. 15, so the discrepancy observed for finite integration time steps is due to the sampling.

4.2. *MFPT in a bistable system*

Given that the contribution to the MFPT which goes like $\sqrt{h}$ is due to the finiteness in the sampling of the stochastic trajectory, we expect that a similar contribution will show up in any MFPT calculation. We plot in Fig. 4 the MFPT

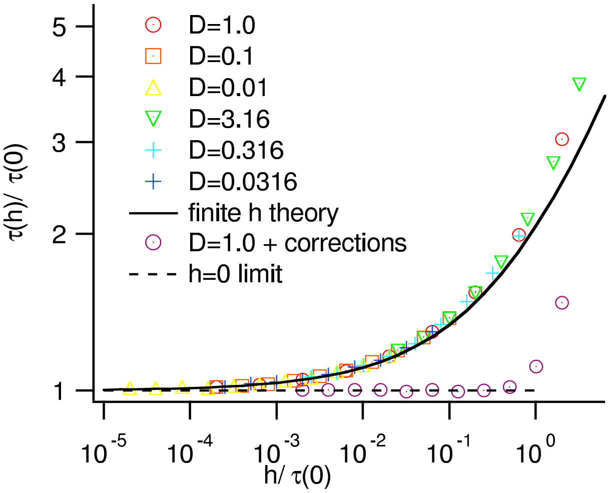

Fig. 3. MFPT for a particle freely diffusing to a boundary

to escape from $x = -1$ to $x = 0$ in the system

$$\dot{x} = x - x^3 + \sqrt{2D}\xi(t), \quad \langle \xi(t) \rangle = 0, \quad \langle \xi(t)\xi(s) \rangle = \delta(t-s). \tag{17}$$

Theoretically, the MFPT in the limit of $h \to 0$ should be

$$\mathrm{MFPT}(h=0) = \frac{\pi}{\sqrt{2}} \exp\left(\frac{1}{4D}\right) \tag{18}$$

From the figure, it is clear that the MFPT's computed without any correction show a square root dependence on the integration time step (best fit). However, when the additional stochastic process to simulate the $h \to 0$ limit is introduced, the numerical points show no dependence on the integration time step. It is important to appreciate that from the structure of Eq. 18 one could believe that, as D becomes smaller, the correction to the MFPT due to the finiteness of h should become negligible compared to the MFPT itself. Fig. 4 shows that the opposite is actually true: MFPT's computed for different h's and D's scale on the same curve, when plotted against $h/\sqrt{D}$: this means that if h is kept constant, the MFPT's simulated without corrections differ, proportionally, more and more from the theoretical ones, as D is made smaller. For more examples, see [12]. The SDE in Eq. 17 was integrated using the Heun algorithm. The statistical error associated with the finite number of trajectories is order of the symbol dimensions.

5. Non white noise

So far we have considered stochastic differential equations driven by white noise. Noise in real systems, however, is very often far from white. An interesting class of

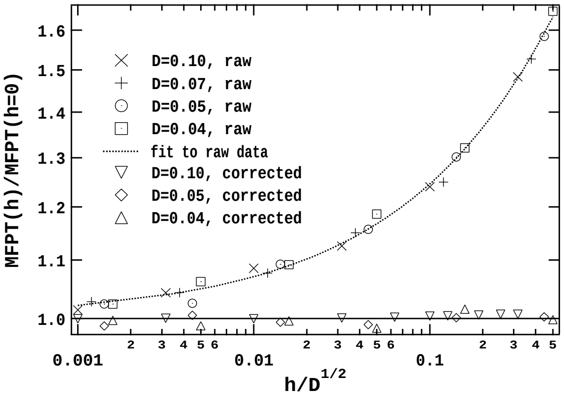

Fig. 4. MFPT in the system of Eq. 17

noise correlations is noise which can be written in terms of linearly filtered white noise. The simplest one of these noises is the exponentially correlated gaussian noise

$$\langle \eta(t) \rangle = 0, \qquad \langle \eta(t)\eta(s) \rangle = \frac{D}{\tau} \exp\left(-\frac{|t-s|}{\tau} \right) \tag{19}$$

or, in term of its spectral density,

$$|\hat{\eta}(\omega)|^2 = \frac{D}{\pi\left(1+\omega^2\tau^2\right)}. \tag{20}$$

The variable $\eta(t)$ can be written in terms of a filtered white noise as follows [14]

$$\dot{\eta} = -\frac{1}{\tau}\eta + \frac{\sqrt{2D}}{\tau}\xi(t), \quad \langle \xi(t) \rangle = 0, \quad \langle \xi(t)\xi(s) \rangle = \delta(t-s) \tag{21}$$

Suppose we have a dynamical system driven by additive exponentially correlated gaussian noise ($\xi(t)$ will be the usual white gaussian noise with standard deviation one)

$$\begin{aligned} \dot{x} &= f(x) + y \\ \dot{y} &= -\frac{1}{\tau}y + \frac{\sqrt{2D}}{\tau}\xi(t). \end{aligned} \tag{22}$$

We could use Eq. 11 (or any other algorithm) to integrate Eq. 22. However, if t_r is the shortest time scale of the dynamical system in Eq. 22, the integration time step h should be chosen so that it is much smaller than both t_r and τ. Now, in cases when τ is much smaller than t_r, it is clear that, making $h \ll \tau$, we would be using all the computing power to integrate the equation describing the exponentially correlated noise, rather than the dynamical system itself.

We recall that Z_1 (Eq. 5) was obtained adding up some gaussian stochastic processes: now, the structure of Eq. 22 is such that y is itself a linear combination of stochastic random gaussian processes ($\xi(t)$), via a filter with time scale τ. So, it should be possible to use the "Full algorithm" or the Heun algorithm, but with a somehow modified $Z_1(h)$. This idea has been exploited in [15]. Eq. 22 can be immediately integrated, to yield (exactly)

$$y(t) = e^{-\frac{t}{\tau}} y(0) + \frac{\sqrt{2D}}{\tau} \int_0^t e^{\frac{s-t}{\tau}} \xi(s)\, ds. \tag{23}$$

Let us now define some quantities which will be needed further down, namely

$$\begin{aligned} w_0 &\equiv \int_0^h e^{\frac{s-h}{\tau}} \xi(s)\, ds \\ w_1 &\equiv \int_0^h \int_0^t e^{\frac{s-t}{\tau}} \xi(s)\, ds\, dt \\ \alpha &\equiv \frac{h}{\tau}. \end{aligned}$$

Clearly, w_0, and w_1 are gaussian variables (they are linear combinations of gaussian variables), with zero average and unknown correlations. Working out the algebra [15], which parallels the algebra carried out to derive Z_1, the result one obtains is summarised in Table 1. From the table, if the correlation of w_0 with w_0 is needed, taking the quantity where the row and the column labelled w_0 cross, one can write

Table 1. Correlations for the stochastic variables in the exponentially correlated algorithm

	$w_0/\tau^{1/2}$	$w_1/\tau^{3/2}$
$w_0/\tau^{1/2}$	$\frac{1}{2}\left(1 - e^{-2\alpha}\right)$	$\frac{1}{2}\left(1 - 2e^{-\alpha} + e^{-2\alpha}\right)$
$w_1/\tau^{3/2}$		$\frac{1}{2}\left(2\alpha - 3 - e^{-2\alpha} + 4e^{-\alpha}\right)$

$$\langle \frac{w_0}{\tau^{1/2}} \frac{w_0}{\tau^{1/2}} \rangle = \frac{1}{2}\left(1 - e^{-2\alpha}\right)$$

which yields

$$\langle w_0^2 \rangle = \frac{\tau}{2}\left(1 - e^{-2\alpha}\right)$$

and so on. We can now write

$$\begin{aligned} y(h) &= y(0) + \frac{\sqrt{2D}}{\tau} w_0 \\ Z_1(h) \equiv \int_0^h y(s)\, ds &= \tau(1 - e^{-\alpha}) y(0) + \frac{\sqrt{2D}}{\tau} w_1 \end{aligned} \tag{24}$$

and, if Y_0 and Y_1 are two independent gaussian variable with zero average and standard deviation one, we can finally write

$$w_0 = \sqrt{\langle w_0^2 \rangle} Y_0$$

$$w_1 = \frac{\langle w_0 w_1 \rangle}{\sqrt{\langle w_0^2 \rangle}} Y_0 + \sqrt{\langle w_1^2 \rangle - \frac{\langle w_1 w_0 \rangle^2}{\langle w_0^2 \rangle}} Y_1.$$

In principle, we need the expressions for $Z_2(h)$ and $Z_3(h)$. $Z_2(h)$ was derived in [15]: however, it turns out that for the exponentially correlated noise case the Heun algorithm is faster than the "Full algorithm", and have a comparable precision. So, it is suggested to use the Heun algorithm, integrating Eqs. 24 to get the $Z_1(h)$ needed at each integration time step.

Another interesting correlated noise is the so caller "green noise" (see [16] for more details and some literature). The spectral density of this noise has the form

$$S(\omega) = \frac{D}{2\pi} \frac{\omega^2}{\omega^2 + \gamma^2}. \tag{25}$$

Algorithms for the numerical integration in the presence of a gaussian noise of arbitrary spectral densities are readily built through a Fourier transform (see, among others, [22,23,4]): the idea is to build the needed noisy process (call it $\xi(t)$) using

$$\xi(t) = \sum_n a_n e^{in\Delta\omega t}. \tag{26}$$

With suitable choices of the coefficients a_n's, any spectral distribution can be generated. It is obvious that the sequence of $\xi(t)$ repeats itself after a time $\bar{t} = 2\pi/(\Delta\omega)$; and that the generation of the sequence is normally handled via an FFT, which means that the whole sequence is a priori generated, having extimated how long it needs to be. In a numerical simulation, this implies that using (26) may be far from optimal. For instance, in case of the passage time to a barrier, one would have to generate a very long random sequence to make sure that there are enough random terms to observe the escape, and yet, once the transition has taken place, what is left of the sequence will have to be rejected. Clearly, this is very inefficient. A much better approach would be to work out an algorithm to generate the noisy sequence which is *local* in time, like shown before for the exponentially correlated noise case. The characteristic of an algorithm which is local in time is that the algorithm requires only the noise value at the previous time step (or at a small and finite number of previous time steps) to generate the noise value for the following time step. The stochastic differential equation which should be integrated has the usual form

$$\dot{x} = a(x) + f(t) \tag{27}$$

where $a(x)$ is a deterministic drift, and $f(t)$ is a stochastic random process, with gaussian statistics, zero average and a spectral density of fluctuations of the form (25). Eq. (25) implies that the moments of $f(t)$, averaged over the noisy realizations of $f(t)$ itself, should be

$$\langle f(t) \rangle = 0 \qquad \langle f(t) f(0) \rangle = D \left[\delta(t) - \frac{\gamma}{2} e^{-\gamma |t|} \right]. \tag{28}$$

In [17] a representation was found for the green noise $f(t)$, in the form

$$f(t) = \xi(t) - \gamma e^{-\gamma t} \int_{-\infty}^{t} e^{\gamma s} \xi(s) ds \tag{29}$$

where $\xi(t)$ is a white gaussian process with moments

$$\langle \xi(t) \rangle = 0 \qquad \langle \xi(t)\xi(0) \rangle = D\delta(t).$$

We could directly integrate these equations: introducing the integration time step h, and defining

$$g(t) = \int_0^t f(s)\, ds, \tag{30}$$

using, for instance, the Heun scheme to step the variable $x(t)$ from $t = 0$ to $t = h$, one would compute

$$\tilde{x} = x(0) + ha(x(0)) + g(h) \qquad x(h) = \frac{1}{2}\left[\tilde{x} + x(0) + ha(\tilde{x}) + g(h)\right]. \tag{31}$$

Although this approach is possible in principle, it is clear that it is very inefficient from a computational point of view. First, we would be forced to store the whole history of the process ξ; second, we would have to compute a lengthy time integral (see Eq. (30)) at each integration time step; and third, this integration would not be local in time. We note, to derive an algorithm which is local in time and paralleling the derivation for exponentially correlated noise, that the (stochastic) variable $g(t)$ is a linear combination of stochastic variables, so that we could approach the problem in a different way: instead of using the stochastic process $g(t)$, defined in (30), we could replace it with an appropriate surrogate stochastic process, characterized by the same statistics and correlation functions.

Let us introduce the quantities

$$I(t') = e^{-\gamma t'} \int_{-\infty}^{t'} e^{\gamma s} \xi(s) ds \tag{32}$$

and the stochastic integrals

$$\begin{aligned} Z_0(t) &= \int_0^t \xi(s)\, ds \\ W_0(t) &= \int_0^t e^{\gamma(s-t)} \xi(s)\, ds \\ W_1(t) &= \int_0^t W_0(s)\, ds \end{aligned} \tag{33}$$

It follows that

$$\begin{aligned} I(t') &= e^{-\gamma t'} \int_{-\infty}^{t'} e^{\gamma s} \xi(s)\, ds = e^{-\gamma t'} \left(\int_{-\infty}^{0} e^{\gamma s} \xi(s)\, ds + \int_0^{t'} e^{\gamma s} \xi(s)\, ds \right) \\ &= e^{-\gamma t'} I(0) + W_0(t'). \end{aligned} \tag{34}$$

Recalling that in the Heun step we need to evaluate $g(h)$, we start rewriting

$$\begin{aligned} g(t) &= \int_0^t f(t')\,dt' = \int_0^t \xi(t')\,dt' - \gamma \int_0^t \left(e^{-\gamma t'} \int_{-\infty}^{t'} e^{\gamma s} \xi(s)\,ds \right) dt' \\ &= Z_0(t) - \gamma \int_0^t \left(e^{-\gamma t'} I(0) + W_0(t') \right) dt' \end{aligned} \tag{35}$$

By integration, we readily obtain

$$g(t) = Z_0(t) + \left(e^{-\gamma t} - 1\right) I(0) - \gamma W_1(t) \tag{36}$$

It should be clear how to carry out the integration: at each time step one uses (34) and (36) to generate the appropriate random variables, which are in turn used in (31) to step the equation forward. At each time step, the previously computed $x(h)$ and $I(h)$ become the new $x(0)$ and $I(0)$, respectively, and so on.

It should be appreciated that the memory of the stochastic process is basically restricted to the term $I(0)$: the quantities $W_0(t')$ and $W_1(t)$ are independent of the previous history of the stochastic process. The whole problem is to find a suitable representation of the stochastic integrals in (34), because clearly the various quantities are *not* independent of each other.

To find a representation of the stochastic integrals (34), we note first that they are linear combinations of gaussian variables, hence they can be represented via a suitable set of random numbers extracted from gaussian distributions of appropriate averages and standard deviations. This makes the problem much simpler to deal with. We only need to specify the first and second moments of these variables, and we should be able to easily generate them. Let us briefly show how to carry out the calculations, which parallels the similar calculations shown above for the exponentially correlated noise case, looking at the second moment of the variable $Z_0(t)$. We have, using $\langle\rangle$ to indicate averages taken over the noise realizations and recalling the definition of $Z_0(t)$ and the statistical properties of the process $\xi(t)$,

$$\langle Z_0(t)^2 \rangle = \langle \int_0^t \int_0^t \xi(s)\xi(s')\,ds\,ds' \rangle = \int_0^t \int_0^t D\delta(s-s')\,ds\,ds' = Dt$$

. It is clear by inspection that all stochastic integrals appearing in (34) have zero average. Their second moments are (we will omit to write the time dependence to keep the notation simple, and assume that $t = h$)

$$\langle Z_0^2 \rangle = Dh$$

$$\langle W_0^2 \rangle = \frac{D}{2\gamma}\left[1 - e^{-2\gamma h}\right]$$

$$\langle W_1^2 \rangle = \frac{D}{2\gamma^3}\left[2\gamma h - 3 + 4e^{-\gamma h} - e^{-2\gamma h}\right]$$

$$\langle Z_0 W_0 \rangle = \frac{D}{\gamma}\left[1 - e^{-\gamma h}\right]$$

$$\langle Z_0 W_1 \rangle = \frac{D}{\gamma^2}\left[\gamma h - 1 + e^{-\gamma h}\right]$$

$$\langle W_0 W_1 \rangle = \frac{D}{2\gamma^2}\left[1 - 2e^{-\gamma h} + e^{-2\gamma h}\right] \tag{37}$$

Suppose that Y_0, Y_1 and Y_2 are random gaussian variables of zero averages and standard deviation one, independent of each other, we can represent the stochastic integrals of (34) as

$$\begin{aligned} Z_0(h) &= \sqrt{\langle Z_0^2 \rangle} Y_0 \\ W_0(h) &= b_0 Y_0 + b_1 Y_1 \\ W_1(h) &= c_0 Y_0 + c_1 Y_1 + c_2 Y_2. \end{aligned} \tag{38}$$

The coefficients appearing in (38) are easily obtained by combining the various quantities appearing in (38), with the help of (37): for example, if we wanted to work out the value for b_0, we would consider

$$\langle W_0(h) Z_0(h) \rangle = \frac{D}{\gamma}\left[1 - e^{-\gamma h}\right] = \langle (b_0 Y_0 + b_1 Y_1) \sqrt{\langle Z_0^2 \rangle} Y_0 \rangle = b_0 \sqrt{\langle Z_0^2 \rangle}$$

and so on.

The analytic expressions for the constants appearing in (38) are cumbersome, and we will not write them here: it is clearly very easy to numerically compute them, given (38) and (37).

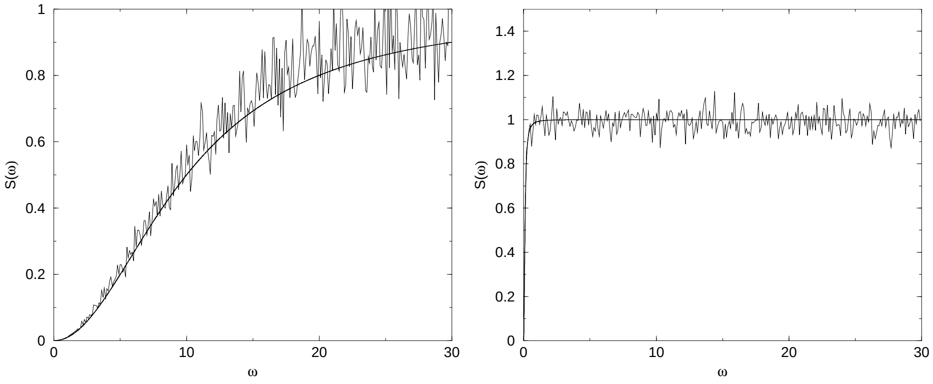

Fig. 5. Spectral density of fluctuations for the quantity $\dot{g}(t) = f(t)$, obtained using Eq. (36), for $D = 2\pi$ and $\gamma = 10.0$ (left), and 0.1 (right). Smooth line: theory (Eq. (25)), jagged line: simulated spectral densities. Note that the agreement between simulation and theory at small frequencies is very good, and that the simulated spectral density does go to a constant for larger frequencies.

Fig. 5 shows that the noise generated using the proposed algorithm has the spectral density (25). Eq. (25) refers to $f(t)$, whereas any integration scheme (like the Heun scheme) requires $g(t)$. Hence, we really generated $g(t)$, differentiated it (computing $(g(t+h)-g(t))/h$), and finally compute the spectral density of fluctuations of the time series obtained. In Figure 5 there are no adjustable parameters. The agreement between generated spectral density of fluctuations and theory is extremely good.

Other noise spectral densities were considered in [20,21]. For general algorithms and further comments, see also [4,22,23].

6. Random number generators

It should be born in mind in the integration of a SDE that a good pseudo random number generator is more important than an efficient integration algorithm. So, particular care must be taken when implementing the code, and the literature should be searched for good generators. There are classical algorithms, like the Box-Muller algorithm [24], to obtain gaussian random variables from uniformly distributed generators; an interesting rejection algorithm (which is faster than the Box-Muller) is the Ziggurath algorithm [25]. Algorithms based on adding a number of uniformly distributed random numbers to obtain a gaussian random number via the central limit theorem should be avoided: they are slower than the Box-Muller, and the generated distribution of random numbers shows a clear cutoff in the tails. There are a number of algorithms to generate a uniformly distributed pseudo random number. The state of art seems to be algorithms based on the so called subtract and carry [26,27] or add and carry [28] algorithms: given their characteristics, these algorithms are particularly well suited in the simulation of rare fluctuations.

7. Conclusions

We have discussed the possibility to have integrators for SDE which are able to reproduce the equilibrium properties of a dynamical system to a high accuracy in the integration time step: such integrators are ideal for studying the long times dynamics of phenomena like a large rare fluctuation. We have shown that it is possible to simulate properties of a zero integration time step process, which implies that we are able to determine with high accuracy when a stochastic trajectory reaches a given threshold. Dedicated algorithms can be derived in case of noise which is filtered through a n poles filter or, more generally, given in terms of linear combination of gaussian noises, speeding up the simulation for these special cases. Finally, some indications as to pseudo random number generators suitable for the case at hand have been given.

References

1. D. G. Luchinsky, P. V. E. McClintock and M. I. Dykman, *Rep. Prog. Phys.* **61**, 889–997 (1998).

2. R. Mannella, in*Noise in Nonlinear Dynamical Systems, III: Experiments and Simulations*, F. Moss and P. V. E. McClintock eds (CUP, Cambridge) 189–221 (1989).
3. P. E. Kloeden and E. Platen *Numerical solution of stochastic differential equations* (Springer-Verlag, Berlin, 1992).
4. R. Mannella, in *Supercomputation in nonlinear and disordered systems*, L. Vasquez, F. Tirado and I. Martin eds (World Scientific) 100–129 (1997).
5. J. Garcia-Ojalvo and J. M. Sancho *Noise in Spatially Extended Systems*, (Springer, Berlin, 1999).
6. N. J. Rao, J. D. Borwankar and D. Ramkrishna *SIAM J. Control* **12** 124–139 (1974)
7. R. Mannella, in *Stochastic Processes in Physics, Chemistry and Biology*, J. A. Freun and T. Pöschel eds (Springer-Verlag, Berlin Heidelberg) 353–364 (2000).
8. T. T. Soong, *Random Differential Equations in Science and Engeneering* (Academic Press, 1973); S. N. Ethier and T. G. Kurtz, *Markov processes characterization and convergence*, (Wiley Series in Probability and Matematical Statistics, 1986); L. Arnold, *Stochastic differential equations: theory and applications*, (Wiley Interscience Publication, 1973); I. I. Gikhman and A. V. Skorokhod, *Stochastic Differential Equations*, (Springer, 1972).
9. G. N. Milstein and M. V. Tret'yakov *SIAM J. Num. Anal.* **34** 2142–2176 (1997); G. N. Milstein, *Theory Probability Applied* **30** 750–766 (1985).
10. G. G. Batrouni, G. R. Katz, A. S. Kronfeld, G. P. Lepage, B. Svetitsky and K. G. Wilson *Phys. Rev. D* **32** 2736–2747 (1985).
11. G Tonnarelli, *Proprietà asintotiche di classi di algoritmi per l'integrazione di equazioni differenziali stocastiche*, (Dissertation for Laurea in Fisica, Pisa, 2001).
12. R. Mannella *Phys. Lett. A* **254** 257–262 (1999).
13. W. Strittmatter *University Freiburg preprint*, THEP 87/12 (1987).
14. J. M. Sancho, M. San Miguel, L. S. Katz and J. Gunton *Phys. Rev. A* **26** 1589–1609 (1982).
15. R. Mannella and V. Palleschi *Phys. Rev. A* **40**, 3381–3386 (1989).
16. R. Mannella *Fluctuations and Noise Letters* **1**, L45–L50 (2001).
17. S. A. Guz, I. G. Ruzavin and M. V. Sviridov, *Slow Dynamics in Systems driven by "Green" Noise'*, in reference [18], pages 515–520.
18. D. Abbott and L. Kish eds., *Unsolved Problems of Noise*, AIP Conference Proceedings Volume 511, Woodbury, NY (2000).
19. S. A. Guz and M. V. Sviridov *P*hys. Lett. A **240** 43–49 (1998).
20. L. Schimansky-Geier and Ch. Zülicke *Z. Phys. B* **79**, 451–460 (1990).
21. M. I. Dykman, R. Mannella, P. V. E. McClintock, N. D. Stein, and N. G. Stocks, *Phys. Rev. E* **47**, 3996–4009 (1993)
22. K. Y. R. Billah and M. Shinozuka *Phys. Rev. A* **42**, 7492–7495 (1990).
23. R. Mannella and V. Palleschi *Phys. Rev. A* **46**, 8028–8030 (1992); K. Y. R. Billah and M. Shinozuka *Phys. Rev. A* **46**, 8031–8033 (1992).
24. W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling *Numerical recipes: the Art of Scientific Computing* (CUP, Cambridge, 1986).
25. G. Marsaglia, and W. W. Tsang *SIAM J. Scie. & Stat. Comp.* **5**, 349–359 (1984).
26. C. M. Luscher *Computer Phys. Comm.* **79**, 100–110 (1994).
27. F. James *Computer Phys. Comm.* **79**, 111-114 (1994).
28. D. E. Knuth *The Art of Computer Programming* (Addison-Wesley, New York), volume 2 (1981).