A gentle introduction to the integration of Stochastic Differential Equations

Riccardo Mannella

Dipartimento di Fisica, Università di Pisa, and Istituto Nazionale Fisica della Materia, UdR Pisa, Via Buonarroti 2, 56100 Pisa, ITALY

Abstract. A gentle introduction to the simulation of stochastic differential equations is presented, with particular attention to the simulation of rare fluctuations, a topic of interest in the light of recent theoretical work on optimal paths. The "best algorithm" and some problems connected to the treatment of the boundaries will be discussed.

1 Introduction

A common denominator of the papers contained in this book is the presence of stochastic processes, introduced to model a variety of different physical situations. Unfortunately, the most common situation is that the stochastic model cannot be exactly solved: 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 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; we should be able to stop in a correct way our simulation when the rare fluctuation hits a prescribed boundary in phase space; we should optimise, if possible, our algorithms to situations when the system lacks detailed balance; and, finally, we should have pseudo random number generators fast and very reliable, able to provide us with very long random sequences. We will address all these problems in this paper. For further comments and references, the interested reader can consult, among others, [2–5].

2 The basic algorithm

A stochastic differential equation has the generic form

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

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 case, the equation has the form

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

A formal integration yields

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

Let us 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 $f_t = f_0 + (x(t) - x(0))f_0' + \ldots$. The simple minded lowest order in h seems to be

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

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 $Z_1(h)$? As we mentioned, we only need its average and its standard deviation. Using $\langle ... \rangle$ to indicate statistical averages,

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

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

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 do that, the correct algorithm to first order in h is

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

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 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

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

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)

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

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:

$$x_1 = x(0) + \sqrt{2D}Z_1(h) + f_0 h$$
$$x(h) = x(0) + \sqrt{2D}Z_1(h) + \frac{h}{2}(f_0 + f(x_1))$$

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.

3.1 Deterministic accuracy

One can disregard the stochastic component, and study 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 $O(h^2)$.

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. 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, $\boldsymbol{x}(t) = \boldsymbol{x}(0) + \boldsymbol{F}(\boldsymbol{x}, t)$, and, for instance, find the equilibrium distribution it generates; then, compare it to the real one. It follows, writing the propagator, that [7]

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

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

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

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: $S(h, x) = O(h)$

It is clear that the "Full algorithm" and the Heun scheme are the algorithms which most faithfully reproduce the 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.

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 [8, 9]. Fig. 1 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. 1), 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.

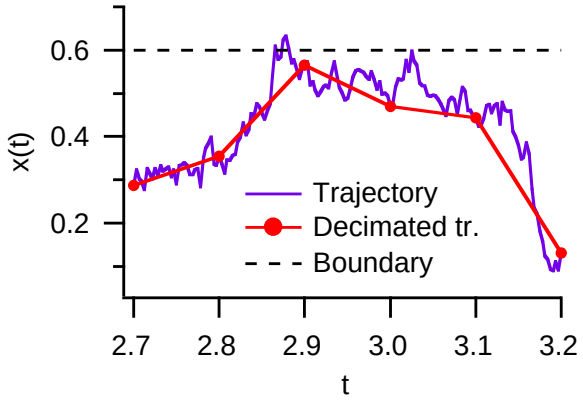

Fig. 1. Comparison between trajectories done with different time steps

Following [8], 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

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

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 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 we 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 [9]. The ruling SDE is

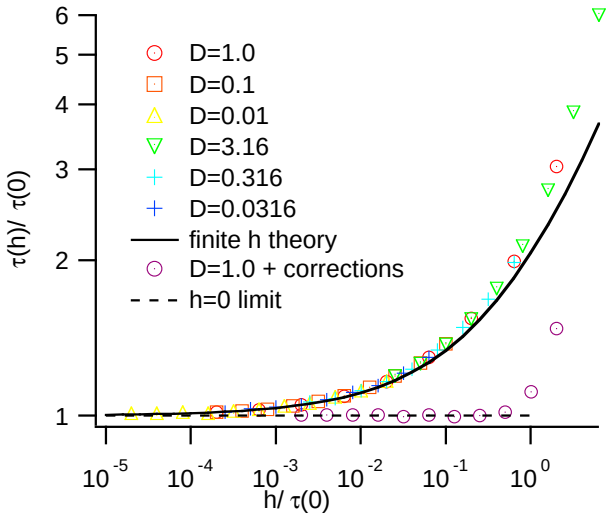

Fig. 2. MFPT for a particle freely diffusing to a boundary

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

For $h = 0$, the MFPT to the boundary is $\tau(0) = \frac{L^2}{2D}$, whereas, for a finite h, it becomes [9]

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

Figure 2 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. 3 the MFPT 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 correc-

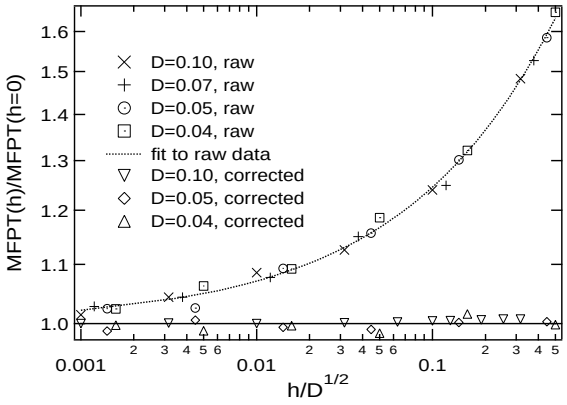

Fig. 3. MFPT in the system of Eq. 17

tion 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. 3 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 [8]. 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 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 [10]

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

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 [11]. 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

$$w_0 \equiv \int_0^h e^{\frac{s-h}{\tau}} \xi(s)\, ds$$

$$w_1 \equiv \int_0^h \int_0^t e^{\frac{s-t}{\tau}} \xi(s)\, ds\, dt$$

$$\alpha \equiv \frac{h}{\tau}.$$

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 [11], 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

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

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

of w_0 with w_0 is needed, taking the quantity where the row and the column labelled w_0 cross, one can write

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

which yields

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

and so on. We can now write

$$y(h) = y(0) + \frac{\sqrt{2D}}{\tau} w_0$$

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

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 [11]: 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. Other noise spectral densities were considered in [12, 13]. For general algorithms and further comments, see also [4, 14, 15].

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 [16], to obtain gaussian random variables from uniformly distributed generators; an interesting rejection algorithm (which is faster than the Box-Muller) is the Ziggurath algorithm [17]. 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 [18, 19] or add and carry [20] algorithms: given their characteristics, these algorithms are particularly well suited in the simulation of rare fluctuations.

7 Conclusions

We have shown that it is possible 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 to study 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, 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. Luchinsky, D. G., McClintock, P. V. E., Dykman, M. I. (1998) Analogue studies of nonlinear systems. Rep. Prog. Phys. **61**, 889–997.

2. Mannella, R. (1989) Computer experiments in nonlinear stochastic physics. *Noise in Nonlinear Dynamical Systems, III: Experiments and Simulations*, F Moss and P V E McClintock eds (CUP, Cambridge) 189–221.
3. Kloeden, P. E., Platen, E. (1992) Numerical solution of stochastic differential equations (Springer-Verlag, Berlin).
4. Mannella, R. (1997) Numerical Integration of stochastic differential equations. *Supercomputation in nonlinear and disordered systems*, Vasquez L, Tirado F and Martin I eds (World Scientific) 100–129.
5. Garcia-Ojalvo, J., Sancho, J. (1999) Noise in Spatially Extended Systems (Springer, Berlin)
6. Rao, N. J., Borwankar, J. D., Ramkrishna, D. (1974) Numerical solution of Ito integral equation. SIAM J. Control **12** 124–139.
7. Batrouni, G. G, Katz, G. R., Kronfeld, A. S., Lepage, G. P., Svetitsky, B., Wilson, K. G. (1985) Langevin simulations of lattice field theories. Phys. Rev. D **32** 2736–2747.
8. Mannella, R. (1999) Absorbing boundaries and optimal stopping in a stochastic differential equation. Phys. Lett. A **254**, 257–262.
9. Strittmatter W. (1987) Numerical Simulation of the Mean First Passage Time. *University Freiburg preprint*, THEP 87/12.
10. Sancho, J. M., San Miguel, M., Katz, L. S., Gunton, J. (1982) Analytical and numerical studies of multiplicative noise. Phys. Rev. A **26**, 1589–1609.
11. Mannella, R., Palleschi, V. (1989) Fast and precise algorithm for computer simulation of stochastic differential equations. Phys. Rev. A **40**, 3381–3386.
12. Schimansky-Geier, L., Zülicke, Ch. (1990) Harmonic noise: Effect on bistable systems Z. Phys. B **79**, 451–460.
13. Dykman, M. I., Mannella, R., McClintock, P. V. E., Stein, N. D., Stocks, N.G. (1993) Probability distributions and escape rates for systems driven by quasi-monochromatic noise. Phys. Rev. E **47**, 3996–4009.
14. Billah, K. Y. R., Shinozuka, M. (1990) Numerical Methods for colored-noise generation and its application to a bistable system. Phys. Rev. A **42**, 7492–7495.
15. Mannella, R., Palleschi, V. (1992) Comment on: "Numerical methods for colored-noise generation and its application to a bistable system". Phys Rev A **46**, 8028–8030; Billah, K. Y. R., Shinozuka, M. (1992) Reply. Phys. Rev. A **46**, 8031–8033.
16. Press, W. H., Flannery, B. P., Teukolsky, S. A., Vetterling, W. T. (1986) Numerical recipes: the Art of Scientific Computing (CUP, Cambridge).
17. Marsaglia, G., Tsang, W. W. (1984) A fast, easily implemented method for sampling from decreasing or symmetric unimodal density functions. SIAM J. Scie. & Stat. Comp. **5**, 349–359.
18. Luscher, C. M. (1994) A portable high-quality random number generator for lattice field theory simulations. Computer Phys. Comm. **79**, 100–110.
19. James, F. (1994) RANLUX: A Fortran implementation of the high-quality pseudorandom number generator of Luscher. Computer Phys. Comm. **79**, 111-114.
20. Knuth, D. E. (1981) The Art of Computer Programming (Addison-Wesley, New York), volume 2.