19 April 1999

PHYSICS LETTERS A

ELSEVIER

Physics Letters A 254 (1999) 257–262

Absorbing boundaries and optimal stopping in a stochastic differential equation

R. Mannella

Dipartimento di Fisica, Università di Pisa, and Istituto Nazionale Fisica della Materia, UdR Pisa, Piazza Torricelli 2, 56100 Pisa, Italy

Received 16 November 1998; revised manuscript received 10 February 1999; accepted for publication 10 February 1999
Communicated by C.R. Doering

Abstract

The problem of optimal stopping to an absorbing boundary in a stochastic differential equation (and more generally, in a stochastic series) is studied. It is shown that a simple minded approach may lead to incorrect results and that some corrections are needed. An algorithm for the correction is introduced and compared to numerical simulations for some case examples. © 1999 Elsevier Science B.V.

PACS: 05.40.+j; 02.50.−r
Keywords: Stochastic processes; Absorbing boundaries

1. Motivation

The numerical integration of some relevant stochastic differential equation (SDE) has become a standard tool in the simulation of the behaviour of real systems. Powerful algorithms have been introduced, and special cases when the stochastic process which drives the differential equation is not white have been studied at some length (for a review, see [1]).

There is however an aspect which has received little attention within the physics community, despite the importance in most of the works published so far. The problem has to do with the finiteness of the time sampling of a generic stochastic trajectory: as such, it is present even in a real situation, because of the finite bandwidth of any device used to gather data from the experiment. In a digital simulation this finite bandwidth comes around because of the finiteness on the integration time step. Fig. 1 clarifies the problem: it is a toy example, obtained integrating the equation

$$\dot{x} = \xi(t)\,, \tag{1}$$

where $\xi(t)$ is a Gaussian process with zero average and correlation $\langle \xi(t)\xi(s)\rangle = 2D\delta(t-s)$. In the numerical simulations, a very small integration time step was used, and then the decimated trajectories, obtained taking points on the trajectory which are separated by larger time steps, were plotted. The problem is immediately apparent: if one asked the time at which the trajectory crosses a given threshold (in the figure, the dashed horizontal line), the trajectory sampled at a larger time interval would miss the hit around $t = 2.86\ldots$, which instead would be visible on the trajectory sampled at a smaller time interval. In this specific case, one could always go back to the "original" trajectory, obtained with a tiny time step: but in most real situations, when perhaps the physical phenomenon evolves on a very long time scale, it is simply not feasible to take a very small time step.

Obviously, unless some trick is employed, the cal-

PII S0375-9601(99)00117-6

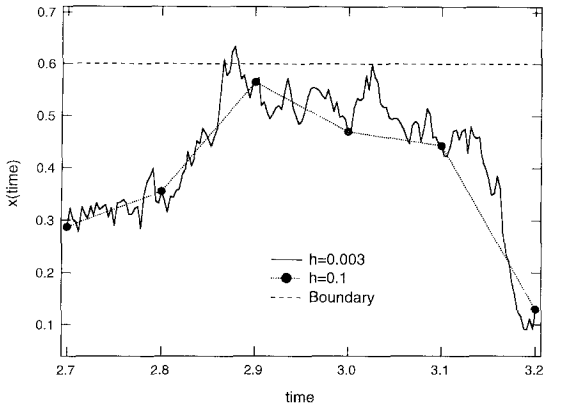

Fig. 1. A typical stochastic trajectory obtained integrating Eq. (1). The trajectories shown were obtain after sampling at two different time intervals a stochastic realization obtained integrating with a time step of 10^{-6}.

culation of a quantity like the Mean First Passage Time (MFPT) to a (herein, absorbing) boundary has a meaning only in the limit in which the sampling time goes to zero, a situation clearly not achievable in a simulation. In the following, we will concentrate on the numerical integration of a SDE: however, it should be kept in mind that these considerations carry over to a generic experimental stochastic series, replacing "integration time step" with "sampling time interval". If a SDE like Eq. (1) is studied, typically one finds that the error associated with the finiteness of the integration time step modifies the observed probability distributions near the boundaries (there is, generally speaking, an excess probability, Fig. 2, top) and the numerical MFPT (which goes like $\sqrt{h}$ where h is the integration time used in the simulations, Fig. 2, bottom). The simulations are relative to a SDE in the form of Eq. (1), where particles are injected at $x = 0$, with absorbing boundary conditions at $x_b = \pm L$. A similar effect is also observed in the presence of a deterministic force on the r.h.s. of Eq. (1): for example, Fig. 3 (top) shows the MFPT, obtained in the SDE,

$$\dot{x} = x - x^3 + \xi(t)\,, \tag{2}$$

where $\xi(t)$ is defined as in Eq. (1), to three different boundaries. It is clear that even in this case the correction to the true MFPT due to the finiteness of the integration time step is not negligible, and it should be taken care of.

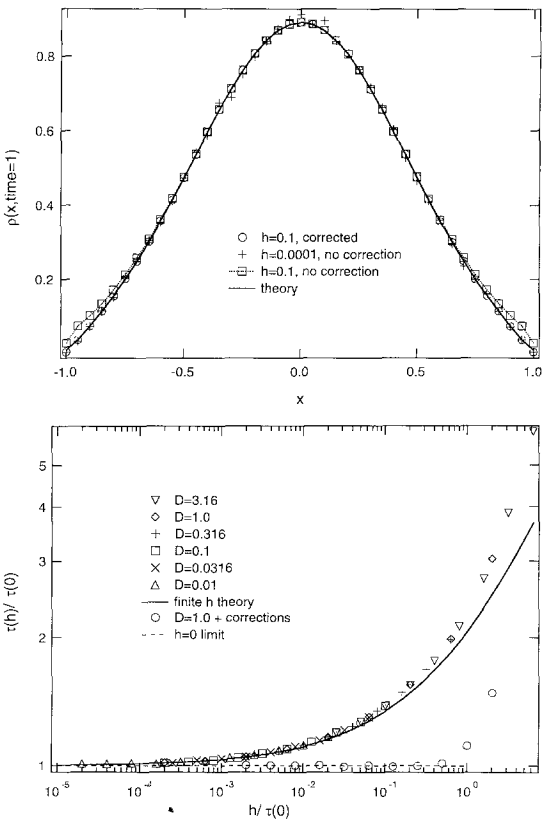

Fig. 2. Simulation results from the integration of Eq. (1), with particles injected at $x = 0$, and absorbing boundaries located at $x_b = \pm L$, $L = 1$; the statistical error is typically in the order of or less than the symbols size. Top: probability distributions at times $t = 1$. Bottom: MFPT to the boundaries. When nothing is indicated, no correction was used.

2. The correction

We will show that the introduction of an auxiliary stochastic process can improve the simulations and somehow overcome the problems shown in the previous section. This auxiliary stochastic process will control the probability that within the finite integration time step h the particle hit the boundary and came back, and it is related to where the particle is (in a statistical sense) at intermediate times.

First, we consider the case of free diffusion, i.e. Eq. (1). From Eq. (1) we have

$$x_h = x_0 + \sqrt{2Dh}\, w_1\,, \tag{3}$$

where w_1 is a Gaussian random variable with stan-

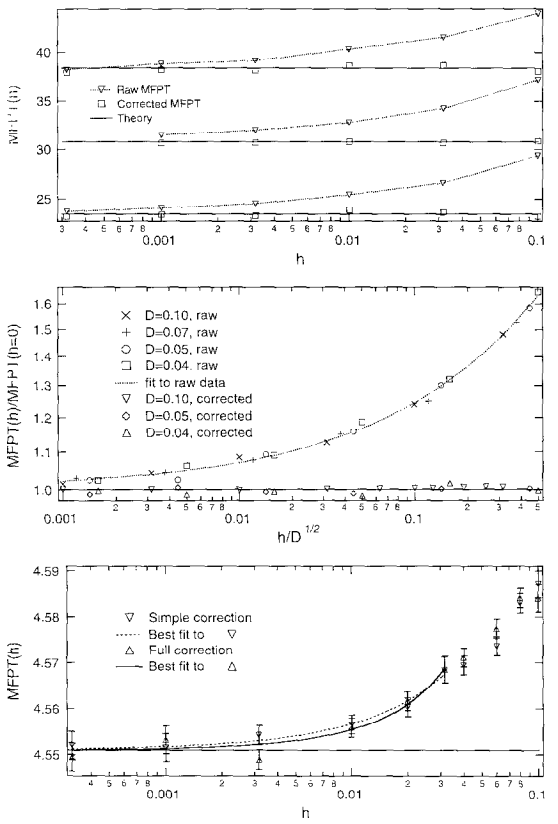

Fig. 3. Simulation results from the integration of Eq. (2), with particles injected at $x = -1$; the statistical error, when not otherwise indicated by an error bar, is smaller than the symbols used in the graphs. Top: MFPT to the boundaries $x_b = 0.3$, $x_b = 0$, $x_b = -0.3$, with and without correction, for different h's and $D = 0.1$. Middle: MFPT to the boundary $x_b = -0.5$, with and without corrections, for different h's and D's; the fit is done with a constant plus a square root of the argument $h/D^{1/2}$. Bottom: MFPT to the boundary $x_b = -0.5$, using Eq. (7) or Eq. (13) to obtain the correction, for different h's and $D = 0.1$.

dard deviation one and average zero (note that given the structure of Eq. (1), Eq. (3) is exact), x_0 is the position at the time $t = 0$ and x_h is the position at the time $t = h$. Next, we need to find where the trajectory is at intermediate times, in a statistical sense. From Eq. (1), at a time zh $(0 < z < 1)$,

$$x(zh) = x_0 + \sqrt{2Dh}(zw_1 + \sqrt{z(1-z)}\,w_2)\,, \quad (4)$$

where w_2 is another Gaussian variable with standard deviation one and average zero, independent of w_1. The stochastic variable w_1 appearing in Eq. (4) is, on the other hand, the same stochastic variable appearing in Eq. (3). With the help of Eq. (3), Eq. (4) becomes

$$x(zh) = x_0 + z(x_h - x_0) + \sqrt{2Dhz(1-z)}\,w_2\,. \quad (5)$$

Eq. (5) is known as Brownian bridge [2] in the literature.

We need to evaluate the probability that at times between 0 and h the trajectory $x(zh)$ exceeded some boundary x_b (without loss of generality, we can assume that $x_0, x_h < x_b$). We first evaluate the probability that the trajectory did not hit the boundary x_b: from [2], this probability is [1]

$$P(\text{no hit}) = 1 - \exp\left[-\frac{(x_h + x_0 - 2x_b)^2}{4Dh} + \frac{(x_h - x_0)^2}{4Dh}\right]. \quad (6)$$

The probability that the trajectory hit the boundary is $1 - P(\text{no hit})$. After straightforward algebra, we have that

$$P(\text{hit}) = \exp\left[-\frac{(x_b - x_0)(x_b - x_h)}{Dh}\right]. \quad (7)$$

Eq. (7) is used as follows: we integrate Eq. (1) using, for instance, the integrator given by Eq. (3). At each integration step, once x_h has been computed, we plug x_0 and x_h into Eq. (7) and evaluate it. If it is larger than a random variable extracted from a flat distribution over $(0, 1)$, we assume that the trajectory did indeed hit the boundary within the integration time step and came back. Hence, the appropriate action for reaching the boundary should be taken: for instance, if we were looking at the MFPT to the boundary, we should stop the integration and store the time since the trajectory was started; if we were looking at the time dependent probability distribution, we should ignore the point x_h in building the distribution, and restart/reinject the trajectory at some initial point, etc.

For a SDE in the form of Eq. (1), an alternative derivation of Eq. (7), using a very elegant trick, was introduced in [3]. We will briefly recall this result because it will be useful both for stopping the numerical integration at times within the integration time step h and for extending Eq. (7) to the nonlinear case. The

[1] Page 63 and Eq. (1.1.8) in [2].

idea is to assume that there are fictitious particles beyond the boundary; and that when a trajectory moves from x_0 to x_h, there is a finite probability that the trajectory which is at x_h at times h belongs to one of the fictitious particles, whereas the "real" particle moved beyond the boundary. A way to look at this trick, in order to quantify it, is to ask the probability that within h a trajectory jumped from x_0 to x_b and back from x_b to x_h: this probability is obviously connected to the probability that a jump of size $(x_b - x_0) + (x_b - x_h)$ took place, in the presence of a stochastic process given by Eq. (3) (see Eq. (1.0.6) in [2] for an expression of this probability). The probability that a particle managed to reach the boundary will be given by the probability associated to the jump which hit the boundary, divided by the probability that the particle, in the actual simulation, jumped from x_0 to x_h (or, in other words, the probability of the random variable w_1 which was used to carry out the current integration step). The algebra yields

$$P(\text{hit}) = \frac{\exp\left[-(2x_b - x_0 - x_h)^2/(4Dh)\right]}{\exp\left[-(x_0 - x_h)^2/(4Dh)\right]} = \exp\left[-\frac{(x_b - x_0)(x_b - x_h)}{Dh}\right], \tag{8}$$

which coincides with Eq. (7).

As mentioned, Eq. (7) gives the probability that we hit the boundary within h and came back. However, as far as the MFPT is concerned, there is, in principle, a problem: the time we compute is discretised, in steps of h, whereas the first hitting time in a real system would be continuous. Although this is likely to lead to a marginal correction in all situations when the MFPT is much larger than h, it would be more elegant to be able to evaluate, in a statistical sense, at what time *within* h the trajectory hit the boundary. First, let us consider a trajectory which crossed the boundary: in a statistical sense, the "exact" time at which the boundary was crossed can be easily evaluated solving Eq. (5) for z, assuming $x(zh) = x_b$. Of course, before we solve Eq. (5) we need to generate a random w_2, thus leading to a "statistical" evaluation of the exact hitting time. When the trajectory was stopped because the criterion involving Eq. (7) was met, we use again a Brownian bridge and estimate the intermediate time at which we hit x_b but, assuming $x_0, x_h < x_b$, we pin the bridge at $x_b + (x_b - x_0)$ for $t = 0$ and at x_h for $t = h$ (in the spirit of the trick which led to Eq. (8)). In other words, we solve for z the equation

$$x_b = (x_b + (x_b - x_0)) + z(x_h + x_0 - 2x_b) + \sqrt{2Dhz(1-z)}\,w_2, \tag{9}$$

where again w_2 is randomly generated prior to finding the solution.

We turn now to the more relevant problem of the motion in the presence of a deterministic force, i.e. of a SDE of the form

$$\dot{x} = F(x) + \xi\,. \tag{10}$$

In deriving Eq. (7), we were facilitated by the simple structure of the propagator, i.e. Eq. (3). For the nonlinear case, however, the propagator has, in general, a much more complex structure. The problem of the derivation of an algorithm to integrate a SDE in the presence of a nonlinear drift has received considerable attention in the literature (see for instance [1,4] and references therein), and we will not review it here: let us simply assume that Eq. (10) is propagated in time with a suitable high order algorithm. We are not interested here in an algorithm for the numerical integration, but rather in a manageable form of $P(\text{hit})$ for the nonlinear case, a $P(\text{hit})$ which should be easily and cheaply evaluated at each integration time step. Any reasonable high order algorithm for the integration of a SDE is a nonlinear function of both x_0 and x_h, and the analytical manipulation to find the equivalent $P(\text{hit})$ starting from such a high order algorithm becomes very cumbersome. However, in most physically interesting situations, and in particular in those cases when we look at the activation to a boundary, the probability that the trajectory crossed within the integration time step is significantly different from zero only in a region near the boundary x_b; this probability will decay exponentially as we move away from x_b. Hence, as far as the derivation of $P(\text{hit})$ is concerned, we use a linearized approximation of Eq. (10) around x_b instead of starting from a high order algorithm. Defining $\epsilon(t) \equiv x(t) - x_b$, $F_b = F(x_b)$, $F_b' = F'(x_b)$, etc., Eq. (10) yields

$$\dot{\epsilon}(t) = F_b + F_b'\epsilon(t) + \xi(t)\,, \tag{11}$$

which can be easily integrated, with the result

$$\epsilon(t) = \epsilon(0)\,\mathrm{e}^{-F_b' t} + \frac{F_b}{F_b'}\left(1 - \mathrm{e}^{-F_b' t}\right) + w_1\sqrt{\frac{2D}{2F_b'}\left[1 - \exp\left(-2F_b' t\right)\right]}, \tag{12}$$

where w_1 is a Gaussian stochastic variable with standard deviation one and average zero. Now, we use the trick of fictitious particles beyond the boundary which we introduced before, and evaluate the probability of a jump which brings the particle to the boundary and back to the final point x_h, etc. The final result reads

$$F(\text{hit}) = \exp\left[-\frac{F_b'}{2D\left(\mathrm{e}^{2hF_b'} - 1\right)} \times \left(x_h - x_b + (x_0 - x_b)\,\mathrm{e}^{hF_b'} - \frac{F_b}{F_b'}\right)^2 + \frac{1}{4Dh}\left(x_1 - \left(x_0 + h\frac{F_0 + F_h}{2}\right)\right)^2\right]. \tag{13}$$

When $F \equiv F' \equiv 0$, we recover Eq. (7); it is obvious that the same procedure previously described for the integration of a SDE is now applied here, using Eq. (13) instead of Eq. (7), alongside with a high order integrator [1,4] to propagate the equations of motion. The main difference with the linear case is that the check implicit in Eq. (13) is done only when the trajectory is inside some suitable region near the boundary, within which both the approximation given by Eq. (11) is reasonable and Eq. (13) is significantly different from zero.

3. Simulations and discussion

Fig. 2 summarizes the results for the simulations of Eq. (1), where particles are injected at $x = 0$, and absorbing boundaries are located at $x_b = \pm L$ with $L = 1$. For the random number generator, see [1]. On the top figure, the probability distribution (which is a non-equilibrium one) for the system computed at $t = 1.0$ is shown: theoretically, this probability distribution should go like

$$P(x,t) = \sum_0^\infty \frac{\cos\left(x\pi(2n+1)/2L\right)}{L}\,\mathrm{e}^{\left(-D(2n+1)^2/L^2\right)t}.$$

As mentioned, without correction an excess probability is expected near the boundaries, because some of the trajectories are not correctly removed. This excess probability is evident in the data done for $h = 0.1$, and it disappears when the integration time step is smaller ($h = 10^{-4}$). Of course, the simulation done for the latter integration time step would require a total CPU time which is 10^3 times larger that the simulation done for the former one. As soon as the corrective stochastic process is introduced, the simulation done with $h = 0.1$ yields a probability distribution which is indistinguishable from the theoretical one. In terms of increase of the CPU time spent in the simulations, this additional stochastic process implies that the CPU time is only a factor slightly smaller two larger than in the case without correction, for the same h value: the gain over the case when the integration time step is made smaller is obvious.

On the bottom figure, the MFPT to the boundaries is shown. Theoretically [3], we have that

$$\tau(0) = \frac{L^2}{2D}$$

and

$$\tau(h)/\tau(0) = 1 + \sqrt{\frac{64Dh}{9\pi L^2}} = 1 + \sqrt{\frac{32}{9\pi}}\sqrt{\frac{h}{\tau(0)}}, \tag{14}$$

where $\tau(h)$ stands for the MFPT expected in the simulations for an integration time step equal to h and in the absence of corrections. Obviously, we would like to obtain $\tau(0)$ in an ideal simulation, whatever the integration time step used. The data obtained from the simulations, using different noise intensities and integration time steps, follow closely Eq. (14). From the figure, even for very small $h/\tau(0)$ the discrepancy is clear. On the other hand, as soon as the correction given by Eq. (7) is introduced, the MFPT from the simulations agrees with $\tau(0)$ up to fairly large $h/\tau(0)$.

In Fig. 3 the result of the simulations for the system described by Eq. (2) is shown. In all simulations a bunch of particles is injected at the potential minimum located at $x = -1$; the algorithm used in the integration is the Heun algorithm (for more details see [1]). In the top graph of Fig. 3, the MFPT's to the boundaries $x_b = -0.3$, $x_b = 0$ and $x_b = 0.3$ are shown: for all boundaries considered, it is clear that the MFPT

computed in the absence of corrections (raw MFPT in the figure) differs substantially from the theoretical one. On the other hand, the corrected MFPT, obtained using Eq. (13), is fairly close to the theoretical one, even for very large integration time steps.

From Eq. (14), in the case of free diffusion the MFPT from the simulations in the absence of corrections differs by the ideal one by a term which goes like $\sqrt{Dh}$. On the other hand, for the system given by Eq. (2), it is known [5] that the theoretical MFPT goes like $\exp[1/4D]$, and diverges as $D \to 0$. So, it may be argued that the results shown in the top graph of Fig. 3 are relevant only for relatively large noise intensities; and that as D becomes smaller, the correction due to the finiteness of h should become negligible compared to the MFPT itself. The middle graph of Fig. 3, however, proves that this is not the case. The MFPT to $x_b = -0.5$ is shown on the graph, for a variety of noise intensities and integration time steps. The fit shown on the figure is done using a square root of the argument $h/D^{1/2}$. The graph proves that for this case the MFPT scales following the relation

$$\mathrm{MFPT}(h)/\mathrm{MFPT}(0) - 1 \propto \sqrt{h/D^{1/2}},$$

which implies that the problem will not disappear as $D \to 0$ but, on the contrary, will be enhanced: making D smaller, the quantity $h/D^{1/2}$ increases and this "pushes" the simulated point to the right, amplifying the error associated with a finite h. In fact, as D becomes smaller and no correction is applied, we should decrease h proportionally to $\sqrt{D}$ to have "comparable" data. It is also clear from the graph, finally, that all problems disappear as soon as the correction of Eq. (13) is introduced.

One final question is whether the correction introduced by Eq. (13) is necessary, or, noting that it contains higher orders in h, perhaps the correction introduced by (the simpler) Eq. (7) would suffice. The bottom graph of Fig. 3 shows that the full correction (Eq. (13)) is more appropriate. The graph compares the two different corrections (Eq. (7) is indicated on the graph as "simple correction", Eq. (13) as "full correction"). The MFPT to the boundary $x_b = -0.5$ for $D = 0.1$ is shown: although it is somehow evident from the graph that the data obtained applying the correction given by Eq. (13) are closer to the theoretical value (the horizontal solid line, note in particular the point at $h = 0.003$), the best fit to the two sets of data, using h as the independent variable, yields an unambiguous result: the data obtained applying the correction given by Eq. (13) are compatible with a constant plus a polynomial function, whereas the data obtained applying the correction given by Eq. (7) are better described by a constant plus a square root function. A constant plus a square root fit to the data obtained using the correction given by Eq. (13), on the other hand, yields a poor χ^2 and it is not appropriate for the description of the data themselves.

4. Conclusions

The aim of this Letter is to stress the importance of a proper treatment of a generic stochastic differential equation when absorbing boundaries are present in the problem. The conclusions are that one must introduce some form of correction to the simple dynamics provided by a straight integration of the relevant stochastic differential equations. Previous work, valid for free diffusing systems, is revised, and interpreted in terms of stochastic processes. A recipe to be used in the presence of deterministic forces is then derived and tested against digital simulations.

References

[1] R. Mannella, in: Supercomputation in Nonlinear and Disordered Systems, F.L. Vasquez, F. Tirado, I. Martin, eds. (World Scientific, Singapore, 1997) pp. 100–129, and references therein.

[2] A.N. Borodin, P. Salminen, Handbook of Brownian Motion (Birkhäuser, Basel, 1996).

[3] W. Strittmatter, U. Freiburg, THEP 87/12.

[4] P.E. Kloeden, E. Platen, Numerical Solution of Stochastic Differential Equations (Springer, Berlin, 1992).

[5] H. Kramers, Physica (Utrecht) 7 (1940) 284;
for a recent review on the subject and extensions to more complex systems, see the papers in Noise in Nonlinear Dynamical Systems, Vol. 1, F. Moss, P.V.E. McClintock, eds. (Cambridge Univ. Press, Cambridge, 1989).