Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-12 (2015) 316–321

Computing the Density Function for a Nonlinear Stochastic Delay System

H. S. Bhat [*] R. W. M. A. Madushani [**]

[*] *University of California, Merced, Merced, CA 95343 USA (e-mail: hbhat@ucmerced.edu)*
[**] *University of California, Merced, Merced, CA 95343 USA (e-mail: rmadushani@ucmerced.edu)*

Abstract: We develop a numerical method to compute the density of a specific nonlinear stochastic delay system, with no sampling. This system arises as a switch-type control model for human balance. Numerical tests against the Euler-Maruyama method show that our method is capable of computing accurate solutions. In particular, the method captures the covariance of the solution at the present and delayed times. This is accomplished through the time-evolution of a Gaussian approximation of the joint density at the present and delayed times. Issues of circularity prevent the numerical solution of the Fokker-Planck equation for stochastic delay systems. Our method bypasses these issues and offers one of the first deterministic algorithms to compute the density of a nonlinear stochastic delay system.

Keywords: stochastic differential equations, time delay, nonlinear systems, numerical analysis

1. INTRODUCTION

Research into the human nervous system's internal control mechanism for maintaining upright balance and posture has led to a number of mathematical models (Eurich and Milton, 1996; Cabrera, 2005; Milton et al., 2009; Boulet et al., 2009; Suzuki et al., 2012; Kowalczyk et al., 2014). Such models typically feature nonlinear feedback, time delay(s), and noise. In this work, we focus on a model that incorporates feedback through a switch-type control mechanism; the model we analyze is the $\tau = \tau'$ case of Eq. (1.2) in (Milton, 2011). The model consists of the following nonlinear stochastic delay differential equation (SDDE):

$$dX_t = f(X_{t-\tau})dt + dW_t \tag{1a}$$

$$f(y) = \alpha y + \begin{cases} C & y < -\beta \\ 0 & |y| \leq \beta \\ -C & y > \beta, \end{cases} \tag{1b}$$

where α, β, C, τ are positive constants, and W_t is the Wiener process or standard Brownian motion. In (1a), X_t is the angle of displacement of the body from the line of gravity at time t. We assume that X_t is a given, deterministic function for $-\tau \leq t \leq 0$.

If the time delay τ in (1a) is zero, the resulting equation is a stochastic differential equation (SDE). With a smooth approximation to the drift function f, we can compute the p.d.f. of this SDE by solving the associated Fokker-Planck or Kolmogorov equation, a deterministic partial differential equation. Because the associated Fokker-Plank equation is circular when $\tau \neq 0$, (Longtin, 2009), SDDE are typically solved using Monte Carlo methods, which involve generating large numbers of sample paths.

In this paper we present a numerical method to solve for the p.d.f. of the nonlinear SDDE (1) with no sampling. The method represents the unknown p.d.f. using a discretization on a spatial grid; the p.d.f. that is computed is not constrained to come from a particular family of densities. In particular, though our method makes a Gaussian approximation for a long-range joint density, the marginal density that is computed is not necessarily Gaussian. While we do not establish rigorous convergence of the method here, we do provide evidence that the method is accurate enough for many purposes.

There is no existing method to compute the p.d.f. of a nonlinear SDDE with large delay, without simulating sample paths. The method described in this paper takes a first step in this direction.

2. DERIVATION OF NUMERICAL METHOD

Let $\ell \geq 1$ be an integer and set the time step $h = \tau/\ell$. Let $\{Z_n\}$ be an i.i.d. family of standard (zero mean, unit variance) Gaussian random variables. Then the Euler-Maruyama discretization of (1) is

$$x_{n+1} = x_n + f(x_{n-\ell})h + \sqrt{h}Z_{n+1}. \tag{2}$$

Because we assumed the initial segment $\{X_s : -\tau \leq s \leq 0\}$ is deterministic, the initial conditions $x_{-\ell}, \cdots, x_0$ are known constants. Our approach is to derive from (2) a deterministic evolution equation that yields an approximate density $p(x_n)$. We view (2) as a reasonable starting point since convergence as $h \to 0$ has been established (Buckwar et al., 2008; Gyöngy and Sabanis, 2013).

2.1 Evolution Equation

Let the joint density of $\mathbf{x}_n = (x_n, x_{n-1}, \ldots, x_{n-\ell})$ be denoted by $p(\mathbf{x}_n)$. Then we have

$$p(\mathbf{x}_{n+1}) = \int_{x_{n-\ell}} p(x_{n+1}|\mathbf{x}_n)p(\mathbf{x}_n)dx_{n-\ell}. \tag{3}$$

2405-8963 © 2015, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2015.09.397

From (2), the conditional density $p(x_{n+1}|\mathbf{x}_n)$ is Gaussian with mean $x_n + f(x_{n-\ell})h$ and variance h. This observation shows that $p(x_{n+1}|\mathbf{x}_n) = p(x_{n+1}|x_n, x_{n-\ell})$. In principle, (3) is a self-contained system for evolving the joint density $p(\mathbf{x}_n)$. The joint density $p(\mathbf{x}_n)$ is a scalar function of $\ell + 1$ variables. It is impractical to store and manipulate spatial discretizations of such functions when $\ell \geq 4$, or equivalently, for time steps $h \leq \tau/4$. For this reason, we seek a simpler evolution equation. (As a Monte Carlo approach, one can directly simulate sample paths of (2); here we seek a method that does not involve sampling.)

Our first step is to integrate both sides of (3) with respect to $(x_{n-1}, \ldots, x_{n-\ell+1})$. This yields

$$p(x_{n+1}, x_n) = \int\limits_{x_{n-\ell}} p(x_{n+1}|x_n, x_{n-\ell}) p(x_n, x_{n-\ell}) dx_{n-\ell}. \quad (4)$$

If $p(x_n, x_{n-\ell})$ is known, we can use (4) to compute $p(x_{n+1}, x_n)$. In order to close the loop, we must be able to compute $p(x_{n+1}, x_{n-\ell+1})$. This is explained next.

2.2 Long-Range Probability

We approximate $p(x_n, x_{n-\ell})$ by a two-dimensional Gaussian with mean vector $\boldsymbol{\mu}_{n,n-\ell}$ and covariance matrix $\Sigma_{n,n-\ell}$. While the true joint density is not Gaussian, a Gaussian is the simplest conceivable model that includes the possibility of covariance between the two random variables x_n and $x_{n-\ell}$. In contrast, if we were to approximate $p(x_n, x_{n-\ell}) \approx p(x_n)p(x_{n-\ell})$, although the joint density is allowed to have a non-Gaussian shape, we are essentially declaring x_n and $x_{n-\ell}$ to be independent and hence to have zero covariance. Through extensive tests of approximations of $p(x_n, x_{n-\ell})$, we find that modeling the covariance is essential; models that assume independence or even conditional independence produce errors that are orders of magnitude larger than the Gaussian approximation.

In order to compute $p(x_{n+1}, x_{n-\ell+1})$, we must determine the covariance matrix $\Sigma_{n+1,n-\ell+1}$. It will turn out to be necessary to compute the more general covariance matrix $\Sigma_{n+1,n+1-q}$. Hence we compute

$$\begin{aligned} &E[x_{n+1}x_{n+1-q}] \\ &= \iint x_{n+1}x_{n+1-q}p(x_{n+1}, x_{n+1-q})dx_{n+1}dx_{n+1-q} \\ &= \iint \iint x_{n+1}x_{n+1-q}p(x_{n+1}|x_n, x_{n+1-q}, x_{n-\ell}) \\ &\qquad \times p(x_n, x_{n+1-q}, x_{n-\ell})dx_{n+1}dx_n dx_{n+1-q}dx_{n-\ell}. \end{aligned}$$

Recall that $p(x_{n+1}|x_n, x_{n+1-q}, x_{n-\ell})$ is $p(x_{n+1}|x_n, x_{n-\ell})$, which is Gaussian with known mean and variance. Hence we carry out the integration over x_{n+1} first, and obtain

$$\begin{aligned} &E[x_{n+1}x_{n+1-q}] \\ &= \iiint x_{n+1-q}(x_n + hf(x_{n-\ell})) \\ &\qquad \times p(x_n, x_{n+1-q}, x_{n-\ell})dx_n dx_{n+1-q}dx_{n-\ell} \\ &= \iint x_n x_{n+1-q}p(x_n, x_{n+1-q})dx_n dx_{n+1-q} \\ &+ h\iint x_{n+1-q}f(x_{n-\ell})p(x_{n+1-q}, x_{n-\ell})dx_{n+1-q}dx_{n-\ell} \\ &= E[x_n x_{n+1-q}] + hE[x_{n+1-q}f(x_{n-\ell})]. \end{aligned} \quad (5)$$

Overall, we have shown that

$$\begin{aligned} \mathrm{Cov}[x_{n+1}, x_{n+1-q}] = \mathrm{Cov}&[x_n x_{n+1-q}] \\ &+ (E[x_n] - E[x_{n+1}])E[x_{n+1-q}] \\ &+ hE[x_{n+1-q}f(x_{n-\ell})]. \end{aligned} \quad (6)$$

Substituting $q = \ell$, we see that we cannot compute $\Sigma_{n+1,n-\ell+1}$ without $\Sigma_{n,n-\ell+1}$. To ensure that we have $\Sigma_{n,n-\ell+1}$, we track all joint densities $p(x_n, x_{n+1-q})$ for all $q = 2, \ldots, \ell$. To make this tractable, we assume all of these joint densities are Gaussian with mean $\boldsymbol{\mu}_{n,n+1-q}$ and covariance matrix $\Sigma_{n,n+1-q}$.

We now show that this yields a self-consistent update scheme. Suppose that at time step n, we have used $p(x_n, x_{n-\ell})$ in (4) to compute $p(x_{n+1}, x_n)$. Using quadrature, we can compute the mean and variance of x_{n+1}, which we store. Further suppose we have all of the inputs required to evaluate the right-hand sides of (6) for $q = 2, \ldots, \ell$. This consists of $\{\Sigma_{n,n+1-q}\}_{q=2}^{\ell}$, $\{\Sigma_{n+1-q,n-\ell}\}_{q=2}^{\ell}$, together with the means of $\{x_j\}_{j=n-\ell}^{n+1}$.

With all of this information, we use (6) together with the stored variances of x_{n+1} and x_{n+1-q} to generate and store the new covariance matrix $\Sigma_{n+1,n+1-q}$ for each $q = 2, \ldots, \ell$. The new mean vector $\boldsymbol{\mu}_{n+1,n+1-q}$ is directly determined by the stored means of x_{n+1} and x_{n+1-q}. This is because the vector of means of the marginal densities of $p(x_{n+1}, x_{n+1-q})$ equals the mean vector of the joint density. The $q = \ell$ case of the new mean vector and new covariance matrix gives us enough information to compute $p(x_{n+1}, x_{n+1-\ell})$, enabling the use of (4) at the next time step. Hence we compute $p(x_{n+2}, x_{n+1})$, which in turn yields the expected value of x_{n+2}. This is the only new mean value required to use (6) at the next time step.

To assemble the remaining information required to use (6) at the next time step, we need $\{\Sigma_{n+1,n+2-q}\}_{q=2}^{\ell}\}$ and $\{\Sigma_{n+2-q,n+1-\ell}\}_{q=2}^{\ell}$. In $\{\Sigma_{n+1,n+2-q}\}_{q=2}^{\ell}$, the $q = 2$ term can be computed by applying quadrature to $p(x_{n+1}, x_n)$. The remaining terms in $\{\Sigma_{n+1,n+2-q}\}_{q=2}^{\ell}$ are a subset of the new covariance matrices we generated above. The collection $\{\Sigma_{n+2-q,n+1-\ell}\}_{q=2}^{\ell}$ consists entirely of covariance matrices computed and stored at previous time steps.

2.3 Computation of the Nonlinear Term in (6)

Let us explain in detail how we compute $E[x_{n+1-q}f(x_{n-\ell})]$ for the particular nonlinearity f given by (1b). We proceed as follows, abbreviating $p(x_{n+1-q}, x_{n-\ell})$ as p:

$$\begin{aligned} &E[x_{n+1-q}f(x_{n-\ell})] \\ &= \iint x_{n+1-q}f(x_{n-\ell})\underbrace{p(x_{n+1-q}, x_{n-\ell})}_{p}dx_{n+1-q}dx_{n-\ell} \\ &= \int\int_{x_{n-\ell}=-\infty}^{-\beta} x_{n+1-q}(\alpha x_{n-\ell} + C)pdx_{n+1-q}dx_{n-\ell} \\ &+ \int\int_{x_{n-\ell}=-\beta}^{\beta} x_{n+1-q}\alpha x_{n-\ell}pdx_{n+1-q}dx_{n-\ell} \\ &+ \int\int_{x_{n-\ell}=\beta}^{\infty} x_{n+1-q}(\alpha x_{n-\ell} - C)pdx_{n+1-q}dx_{n-\ell}. \end{aligned}$$

Combining terms with α and C, respectively, we obtain