Zhao, Zheng; Tronarp, Filip; Hostettler, Roland; Särkkä, Simo

State-space Gaussian Process for Drift Estimation in Stochastic Differential Equations

Published in:
Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2020

DOI:
10.1109/ICASSP40776.2020.9054472

Published: 01/05/2020

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:
Zhao, Z., Tronarp, F., Hostettler, R., & Särkkä, S. (2020). State-space Gaussian Process for Drift Estimation in Stochastic Differential Equations. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2020* (pp. 5295-5299). Article 9054472 (Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing). IEEE.
https://doi.org/10.1109/ICASSP40776.2020.9054472

STATE-SPACE GAUSSIAN PROCESS FOR DRIFT ESTIMATION IN STOCHASTIC DIFFERENTIAL EQUATIONS

Zheng Zhao[†]*, Filip Tronarp*[†]*, Roland Hostettler*[⋆]*, and Simo Särkkä*[†]

Department of Electrical Engineering and Automation, Aalto University[†]
Department of Electrical Engineering, Uppsala University[⋆]

ABSTRACT

This paper is concerned with the estimation of unknown drift functions of stochastic differential equations (SDEs) from observations of their sample paths. We propose to formulate this as a non-parametric Gaussian process regression problem and use an Itô–Taylor expansion for approximating the SDE. To address the computational complexity problem of Gaussian process regression, we cast the model in an equivalent state-space representation, such that (non-linear) Kalman filters and smoothers can be used. The benefit of these methods is that computational complexity scales linearly with respect to the number of measurements and hence the method remains tractable also with large amounts of data. The overall complexity of the proposed method is $\mathcal{O}(N \log N)$, where N is the number of measurements, due to the requirement of sorting the input data. We evaluate the performance of the proposed method using simulated data as well as with real-data applications to sunspot activity and electromyography.

***Index Terms*—** Gaussian process regression, Kalman filter and smoother, stochastic differential equation, drift estimation, sunspot activity, electromyography

1. INTRODUCTION

Stochastic differential equations (SDEs) have important applications in areas such as financial marketing [1], astronomy [2], bioinformatics [3,4], and classical physics [5]. SDEs can be seen as ordinary differential equations which involve randomness, and their theory is covered by the Itô stochastic calculus [6]. However, it is not always possible to derive an SDE model of a system directly from first principles and instead, the SDE needs to be estimated from data.

One way to proceed is to fix the parametric form of the SDE and estimate the parameters from data. Such estimation can be done, for example, with Markov chain Monte Carlo methods [7]. Alternatively we can use likelihood-based methods such as maximum likelihood or maximum a posterior [7–9]. Early studies focused on parametrizing with for example, polynomial forms [10–12].

This work was supported by Business Finland

Alternatively, as proposed by [7, 13], we can also model the drift and diffusion functions as Gaussian processes (GPs). This is a non-parametric approach, as we do not need to specifically postulate forms of the functions. Compared to parametric approaches, GP regression provides more flexibility, as the uncertainties can be readily modeled. Current state-of-the-art studies [7, 13–15] model the relationship between measurements and drift and diffusion functions with Euler–Maruyama discretization scheme [16], which results in a linear model, and the posterior of unknown functions is still Gaussian with closed-form mean and covariance [7, 17]. However, one problem of GP regression is that the computational complexity scales cubically with the number of measurements [17]. Although it can be alleviated by utilizing sparse GPs [18], they are only approximations to the full batch GPs. Another problem is the use of Euler–Maruyama, which linearly approximates the Itô–Taylor series in a small time interval. For this reason, the accuracy of the approach is limited when the sample path is not densely discretized [16].

The contributions of this paper are the following. Within the framework of non-parametric GPs for learning unknown drift functions in SDEs, we first propose to generalize the discretization scheme to higher order Itô–Taylor expansions instead of the Euler–Maruyama scheme. This gives more accurate approximations with larger time intervals. Secondly, we transform the GP into an equivalent state-space representation such that the regression of the drift function reduces to a filtering and smoothing problem in state-space [6, 19–21]. When the Kalman filter and Rauch–Tung–Striebel (RTS) smoother are used, the computational complexity is linear in the number of measurements.

2. ITÔ–TAYLOR GAUSSIAN PROCESS REGRESSION OF SDES

Consider the following stochastic differential equation

$$\mathrm{d}x(t) = a(x(t))\ \mathrm{d}t + b\ \mathrm{d}W(t), \tag{1}$$

for the solution process $x_k \triangleq x(t_k) \in \mathbb{R}$, where $W(t)$ is a Wiener process with spectral density q. The aim is to learn the unknown drift function $a_k \triangleq a(x_k)$ from N discrete observations $\mathbf{x}_{1:N} = [x_1, \ldots, x_k, \ldots, x_N]^\mathsf{T}$ of one sample path

of $x(t)$, where time interval $\Delta t_k = t_k - t_{k-1}$. Similarly to [7, 13–15], we model the drift function as a GP, that is,

$$a(x) \sim \mathcal{GP}(0, k_a(x, x')), \tag{2}$$

where $k_a(\cdot, \cdot)$ is a covariance function. We first need to discretize the continuous model (1). Typically, this is done using the Euler–Maruyama approximation of the SDE (1), given by

$$x_{k+1} = x_k + a(x_k)\,\Delta t_k + b\,\Delta W_k, \tag{3}$$

where $\Delta W_k = W(t_{k+1}) - W(t_k) \sim \mathcal{N}(0, q\,\Delta t_k)$. This can now be rewritten as

$$y_k = a(x_k)\,\Delta t_k + b\,\Delta W_k, \tag{4}$$

where we have defined the measurement $y_k = x_{k+1} - x_k$, and x_k for $k = 1, \ldots, N$ are actual observations from sample path. As shown by [13–15] this can now be considered as Gaussian process regression model of the form

$$\begin{aligned} a(x) &\sim \mathcal{GP}(0, k_a(x, x')) \\ y_k &= a(x_k)\,\Delta t_k + \epsilon_k, \end{aligned} \tag{5}$$

with $\epsilon_k \sim \mathcal{N}(0, b^2\,q\,\Delta t_k)$. Thus given the values of x_k and y_k we can use standard Gaussian process regression to infer $a(x)$. The Euler–Maruyama method has the advantage that the resulting posterior is still a GP with closed-form solution [17], but will only work when Δt_k are small enough, as its strong order of convergence is merely $\mathcal{O}(\Delta t^{1/2})$ [6, 16].

Another more accurate approach is to use Itô–Taylor expansion for the SDE (1) in more general form

$$x_{k+1} = x_k + g_k(x_k) + \epsilon_k(x_k), \tag{6}$$

where $g_k(x_k)$ is a deterministic function of x_k and $\epsilon_k(x_k)$ is a random variable. Itô–Taylor methods are methods to approximate the terms g_k and ϵ_k using finite truncations of the Itô–Taylor series expansion. The Euler–Maruyama method is a special case of Itô–Taylor expansion where $g(x_k) = a(x_k)\,\Delta t_k$, $\epsilon_k(x_k) = b\,\Delta W_{t_k}$, and $\Delta W_k \sim \mathcal{N}(0, q\,\Delta t_k)$.

A particularly useful instance is strong order 1.5 Itô–Taylor (Itô-1.5) scheme [16] that uses

$$\begin{aligned} g_k(x_k) &= a(x_k)\,\Delta t_k \\ &\quad + \frac{1}{2}\left(\frac{\mathrm{d}a(x_k)}{\mathrm{d}x_k}\,a(x_k) + \frac{1}{2}\frac{\mathrm{d}^2 a(x_k)}{\mathrm{d}x_k^2}\,b^2\,q\right)\Delta t_k^2, \\ \epsilon_k(x_k) &= b\,\Delta\omega_k + \frac{\mathrm{d}a(x_k)}{\mathrm{d}x_k}\,b\,\Delta\gamma_k, \end{aligned} \tag{7}$$

with

$$\begin{bmatrix} \Delta\gamma_k \\ \Delta\omega_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} q\Delta t_k^3/3 & q\Delta t_k^2/2 \\ q\Delta t_k^2/2 & q\Delta t_k \end{bmatrix}\right). \tag{8}$$

The benefit is that the derivatives of the drift function are also taken into account in regression.

The full formulation for drift function regression using GP and Itô–Taylor expansion then becomes

$$\begin{aligned} a(x) &\sim \mathcal{GP}(0, k_a(x, x')), \\ y_k &= g_k(x_k) + \epsilon_k(x_k), \end{aligned} \tag{9}$$

where the form of $g_k(x_k)$ and $\epsilon_k(x_k)$ depend specifically on how the Itô–Taylor expansion is done, a particular example being Itô-1.5 in (7). Unfortunately, performing GP regression for model (9) is more challenging than using Euler–Maruyama. Firstly, the measurement function in (9) is non-linear and non-Gaussian and hence closed form Gaussian process regression is no longer possible. Secondly, the computational complexity of a vanilla GP is $\mathcal{O}(N^3)$, and we also need the derivatives of $a(x)$.

However, these challenges can be solved by transforming the GP prior into an equivalent state-space model [19, 20] such that the regression is done with (non-linear) filtering and smoothing.

3. STATE-SPACE DRIFT ESTIMATION OF SDE

3.1. State-space Gaussian Process

As proposed by [19, 20], a scalar temporal Gaussian process can be converted to a multidimensional linear stochastic differential equation (i.e., state-space model), provided that the Fourier transform of the GP kernel function has a rational form. The major benefit of using state-space representation of GP is that the computational complexities of Kalman filter and RTS smoother, which can be used to solve the GP regression problem in state-space form, are $\mathcal{O}(N)$. The derivatives of the GP also naturally appear as state components, which can be easily exploited.

However, the inputs of the GP have to be in temporal order to use the state-space representation of GPs, which is not the case in model (4). This can be accomplished by sorting pairs (x_k, y_k) with respect to the values of x_k which leads to new pairs (x_i, y_i). Let us also interpret the inputs x_i as (pseudo) time-variables and denote $\tau_i \triangleq x_i$. Then the GP model (9) can be rewritten as a temporal GP regression model

$$\begin{aligned} a(\tau) &\sim \mathcal{GP}(0, k_a(\tau, \tau')), \\ y_i &= g_i(\tau_i) + \epsilon_i. \end{aligned} \tag{10}$$

If the kernel function k_a is suitably chosen [6], then the spectral density of the GP will have a rational form and we can form a spectral factorization

$$S(\omega) = H(i\,\omega)\,q\,H(-i\,\omega), \tag{11}$$

where $H(i\,\omega)$ is the transfer function that corresponds to a differential equation. This leads to an equivalent linear time-

invariant state-space representation of (10) such that

$$\begin{aligned} \mathrm{d}\boldsymbol{\alpha}(\tau) &= \mathbf{F}\,\boldsymbol{\alpha}(\tau)\,\mathrm{d}\tau + \mathbf{L}\,\mathrm{d}\mathbf{B}(\tau), \\ y_i &= g_i(\tau_i) + \epsilon_i, \end{aligned} \tag{12}$$

where $\boldsymbol{\alpha}(\tau) \in \mathbb{R}^D$ is a D-dimensional state containing $a(\tau)$ and its derivatives and $\mathbf{B}(\tau)$ a Wiener process with a given diffusion matrix $\mathbf{Q} \in \mathbb{R}^{S\times S}$. The initial distribution is given by $\boldsymbol{\alpha}(\tau_0) \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_\infty)$ where τ_0 is some value $\tau_0 < \tau_1$ and $\mathbf{P}_\infty$ is the solution to the corresponding Lyapunov equation [6]. Depending on the specific form of kernel function k_a, the matrices $\mathbf{F}$, $\mathbf{L}$, $\mathbf{Q}$, and $\mathbf{P}_0$ have different forms. A typical example is the Matérn kernel

$$k_a(\tau,\tau') = \sigma^2\,\frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}|\tau-\tau'|}{\ell}\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}|\tau-\tau'|}{\ell}\right),$$

for which we have [6]

$$\mathbf{F} = \begin{bmatrix} 0 & 1 & & \\ & 0 & 1 & \\ \vdots & & \ddots & \\ -\phi_1\lambda^p & -\phi_2\lambda^{D-1} & \cdots & -\phi_D\lambda \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix},$$

where $\lambda = \sqrt{2\nu}/\ell$, $D = \nu + 1/2$, and $\phi_i = \binom{D}{i-1}$ are the binomial coefficients. The details of other parameters $\mathbf{P}_\infty$, $\mathbf{Q}$, and derivations can be found in [6, Chap. 12].

3.2. Filtering and Smoothing

Given the state-space representation (12), and when the measurement model is linear (as it is in the case of Euler–Maruyama), a Kalman filter and RTS smoother can be used to solve the estimation problem in linear computational time [19]. The resulting smoothing posterior at τ

$$p(\boldsymbol{\alpha}(\tau) \mid y_1, \ldots, y_N) = \mathcal{N}(\boldsymbol{\alpha}(\tau); \mathbf{m}^s(\tau), \mathbf{P}^s(\tau)), \tag{13}$$

is exactly the same as of full GP regression, where $\mathbf{m}^s(\tau)$ and $\mathbf{P}^s(\tau)$ are the posterior mean and covariance calculated by the smoother. The posterior distribution of $a(\tau)$ can then be extracted via marginalization.

In the case of adopting higher order Itô–Taylor expansion as constructed in Section 2, the measurement function becomes non-linear, and exact Kalman filtering and smoothing cannot be applied anymore. Instead, we can resort to general Gaussian filtering and smoothing approaches for nonlinear systems such as statistically linearized and sigma-point Kalman filters (e.g., the cubature or unscented Kalman filters) [21, 22]. A particularly useful state-of-the-art tool for this purpose is the iterated posterior linearization filter (IPLF) and smoother (IPLS) [23].

It is worth noting that although the time-complexity of the filters and smoothers is $\mathcal{O}(N)$, the overall time-complexity is dominated by sorting which is typically an $\mathcal{O}(N \log N)$ operation.

3.3. Extension to Diffusion Estimation

It would be also of interests to learn the drift and diffusion functions jointly from a more generalized SDE

$$\mathrm{d}x(t) = a(x(t))\,\mathrm{d}t + b(x(t))\,\mathrm{d}W(t). \tag{14}$$

In this case, we proceed to form state-space models for both a and b and the inference can be done using iterated nonlinear Kalman filters and smoothers. However, the inference problem is now significantly harder, because the process b appears multiplicatively in the model and thus standard nonlinear Kalman filter and smoother solutions can fail.

4. NUMERICAL EXPERIMENTS

4.1. Synthetic Toy Models

We first consider the Ginzburg–Landau double well SDE

$$\mathrm{d}x = 3(x - x^3)\,\mathrm{d}t + \mathrm{d}W, \quad x_0 = 1, \tag{15}$$

and the modified Beneš SDE

$$\mathrm{d}x = \tanh(x)\,\mathrm{d}t + 0.01\,\mathrm{d}W, \quad x_0 = 0, \tag{16}$$

where W is a standard Wiener process. We generate samples from them on fixed time spans of lengths $T = 50$ s and $T = 5$ s for models (15) and (16), respectively. The range and variation of Δt is found in Figure 2. The simulation is done using Euler–Maruyama with small enough sampling interval $\Delta t \times 1\mathrm{e}^{-4}$ s. For calculating the root mean square error (RMSE), we query predictions on fixed dense grids. We use 1000 independent Monte Carlo trials for computing the average RMSE and CPU time.

For the drift function estimation with GPs, we adopt Mátern $\nu = 5/2$ kernel with $\ell = 1.2$, $\sigma = 3$, and $\ell = 1.5$, $\sigma = 0.3$ for models (15) and (16), respectively. The equivalent parameters for state-space model are calculated by (13) and [6]. For comparison, we use Euler–Maruyama scheme for full batch GPs, fully independent conditional (FIC) and deterministic training conditional (DTC) sparse GPs [17], and Kalman filter and RTS smoother (KF-RTS). We also test Itô-1.5 (7) for higher order modeling with unscented Kalman filter (UKF-RTS), IPLF-RTS, and IPLS.

For comparison of full batch GP, sparse GP, and KF-RTS, we show their RMSE and CPU time in Table 1. The estimation results of full GP and KF-RTS are exactly the same, however, the time requirement of KF-RTS is significantly lower. Sparse GPs have similar time performance with KF-RTS, but they have trade-off on estimation accuracy, as they use inducing-points for approximating the solution. The same conclusion is also found in Figure 2.

Demonstration of drift estimation from one Monte Carlo run is shown in Figure 1. The detailed RMSE of all methods for two models is shown in Figure 2. It can be seen that

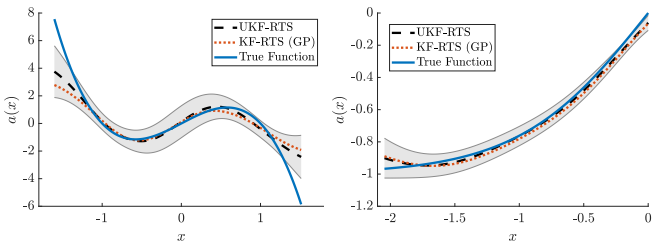

Fig. 1: Demonstration of drift function estimation ($\Delta t = 0.1$ s) for model (15) and (16), respectively. Shaded area stands for 0.95 confidence interval of UKF-RTS.

	RMSE	CPU Time (s)
Kalman	1.42	0.245
GP	1.42	330.034
Sparse GP (FIC)	1.49	0.225
Sparse GP (DTC)	1.49	0.044

Table 1: RMSE and CPU time of drift estimation for model (15) with Euler–Maruyam and $\Delta t = 0.002$ s.

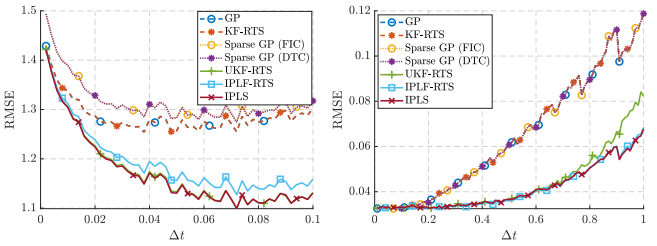

Fig. 2: RMSE of drift estimation for model (15) (left) and (16) (right) as function of Δt.

the methods IPLF, IPLS, and UKF-RTS using non-linear Itô-1.5 outperform the methods that use the Euler–Maruyama scheme and GPs, especially with larger Δt. IPLF and IPLS perform almost identically to UKF-RTS, while they are slightly better for model (16). However, it is also worth noticing that by decreasing Δt, the RMSE for double well model (15) is not improved.

4.2. Real-Data Applications to Sunspot Activity and Electromyography (EMG)

For demonstration of drift function estimation on real-world data, we use Monthly sunspot activity dataset by WDC-SILSO, Royal Observatory of Belgium, Brussels, and Electrocardiogram (ECG) dataset by Aalto university and Helsinki University Central Hospital [24]. We adopt the same estimation methods and Mátern kernel as done in previous toy models with parameters $\ell = 10$, $\sigma = 1.2$, and $\ell = 0.1$, $\sigma = 2$ for sunspot data and EMG data, respectively.

The drift estimation result for sunspot data is shown in

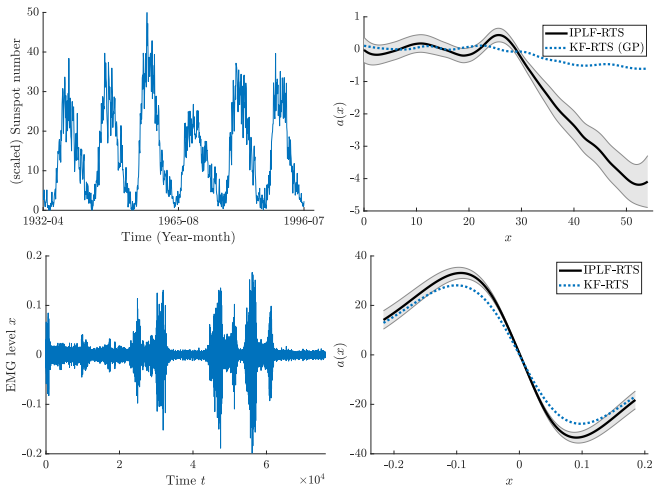

Fig. 3: Drift estimation results (right column) for sunspot (top left) and EMG (bottom left) data. Shaded area stands for 0.95 confidence interval of IPLF-RTS. UKF-RTS and IPLS are not shown as their results are visually indistinguishable from IPLF.

Figure 3. We notice that the GP regression, or equivalently, KF-RTS, with Euler–Maruyama fails to give sensible function estimation, as the result behaves almost as zero-mean random walk. However, the estimation results from IPLF and other methods with Itô-1.5 scheme seem reasonable, because the drift function should have a linear plus bias form [2].

The EMG data is manually extracted from the original ECG dataset by expert. The EMG signal used here contains 77101 sample points and therefore the full batch GP regression fails due to memory limitation, while KF-RTS still works with moderate computational effort. By observing the EMG signal on Figure 3, we find it has significant changes when the magnitude reaches around -0.1 mV and 0.1 mV due to human body motion. This seems to manifest in the drift function estimation, which has peak values around -0.1 mV and 0.1 mV, correspondingly.

5. CONCLUSION

In this paper, we presented a novel state-space Gaussian process regression framework for estimation of unknown drift functions of SDEs. The problem is formulated as a filtering and smoothing problem by transforming the GP into state-space form, such that the computational complexity is linear with respect to the number of measurements. Together with data sorting procedure, the overall averaged complexity of the method is $\mathcal{O}(N \log N)$. We also used an Itô–Taylor expansion for accurate modeling of the unknown drift SDE functions. The estimation performance of the proposed method is shown to be in line with the state-of-the-art by using synthetic and real data experiments.

6. REFERENCES

[1] Damien Lamberton and Bernard Lapeyre, *Introduction to Stochastic Calculus Applied to Finance*, Chapman and Hall/CRC, 2007.

[2] Edward J. Allen and Chisum Huff, "Derivation of stochastic differential equations for sunspot activity," *Astronomy & Astrophysics*, vol. 516, 2010.

[3] Věra Lánská, Petr Lánský, and Charles E. Smith, "Synaptic transmission in a diffusion model for neural activity," *Journal of Theoretical Biology*, vol. 166, no. 4, pp. 393–406, 1994.

[4] Zheng Zhao, Simo Särkkä, and Ali Bahrami Rad, "Spectro-temporal ECG analysis for atrial fibrillation detection," in *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sep. 2018, pp. 1–6.

[5] David M. Ceperley and Berni J. Alder, "Ground state of the electron gas by a stochastic method," *Physical Review Letters*, vol. 45, pp. 566–569, Aug 1980.

[6] Simo Särkkä and Arno Solin, *Applied Stochastic Differential Equations*, Institute of Mathematical Statistics Textbooks. Cambridge University Press, 2019.

[7] Omiros Papaspiliopoulos, Yvo Pokern, Gareth O. Roberts, and Andrew M. Stuart, "Nonparametric estimation of diffusions: a differential equations approach," *Biometrika*, vol. 99, no. 3, pp. 511–531, 07 2012.

[8] Roland Hostettler, Filip Tronarp, and Simo Särkkä, "Modeling the drift function in stochastic differential equations using reduced rank Gaussian processes," in *18th IFAC Symposium on System Identification (SYSID)*, July 2018, pp. 778–783.

[9] Rudolf Friedrich, Joachim Peinke, Muhammad Sahimi, and M. Reza Rahimi Tabar, "Approaching complexity by stochastic methods: From biological systems to turbulence," *Physics Reports*, vol. 506, no. 5, pp. 87–162, 2011.

[10] Mathieu Kessler, "Estimation of an ergodic diffusion from discrete observations," *Scandinavian Journal of Statistics*, vol. 24, no. 2, pp. 211–229, 1997.

[11] Didier Dacunha-Castelle and D. Florens-Zmirou, "Estimation of the coefficients of a diffusion from discrete observations," *Stochastics*, vol. 19, no. 4, pp. 263–284, 1986.

[12] David Lamouroux and Klaus Lehnertz, "Kernel-based regression of drift and diffusion coefficients of stochastic processes," *Physics Letters A*, vol. 373, no. 39, pp. 3507–3512, 2009.

[13] Andreas Ruttor, Philipp Batz, and Manfred Opper, "Approximate Gaussian process inference for the drift function in stochastic differential equations," in *Advances in Neural Information Processing Systems 26*, pp. 2040–2048. MIT Press, 2013.

[14] Constantino A. García, Abraham Otero, Paulo Félix, Jesús Presedo, and David G. Márquez, "Nonparametric estimation of stochastic differential equations with sparse Gaussian processes," *Physical Review E*, vol. 96, Aug 2017.

[15] Philipp Batz, Andreas Ruttor, and Manfred Opper, "Approximate Bayes learning of stochastic differential equations," *Physical Review E*, vol. 98, Aug 2018.

[16] Peter E. Kloeden and Eckhard Platen, *Numerical Solution of Stochastic Differential Equations*, Springer–Verlag Berlin Heidelberg, 1992.

[17] Carl Edward Rasmussen and Christopher K. I. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, 2006.

[18] Edward Snelson and Zoubin Ghahramani, "Sparse Gaussian processes using pseudo-inputs," in *Advances in Neural Information Processing Systems 18*, pp. 1257–1264. MIT Press, 2006.

[19] Jouni Hartikainen and Simo Särkkä, "Kalman filtering and smoothing solutions to temporal Gaussian process regression models," in *2010 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Aug 2010, pp. 379–384.

[20] Simo Särkkä, Arno Solin, and Jouni Hartikainen, "Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing: A look at Gaussian process regression through Kalman filtering," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 51–61, July 2013.

[21] Zheng Zhao, Toni Karvonen, Roland Hostettler, and Simo Särkkä, "Taylor moment expansion for continuous-discrete Gaussian filtering and smoothing," *arXiv preprint arXiv:2001.02466*, 2020.

[22] Simo Särkkä, *Bayesian Filtering and Smoothing*, Institute of Mathematical Statistics Textbooks. Cambridge University Press, 2013.

[23] Ángel F. García-Fernández, Lennart Svensson, and Simo Särkkä, "Iterated posterior linearization smoother," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 2056–2063, April 2017.

[24] Roland Hostettler, Tuomas Lumikari, Lauri Palva, Tuomo Nieminen, and Simo Särkkä, "Motion artifact reduction in ambulatory electrocardiography using inertial measurement units and Kalman filtering," in *2018 21st International Conference on Information Fusion (FUSION)*, July 2018, pp. 1–8.