On Simulation and Trajectory Prediction with Gaussian Process Dynamics

Lukas Hewing[1]   LHEWING@ETHZ.CH

Elena Arcari[1]   EARCARI@ETHZ.CH

Lukas P. Fröhlich[1,2]   LUKASFRO@ETHZ.CH

Melanie N. Zeilinger[1]   MZEILINGER@ETHZ.CH

[1]*Institute for Dynamic Systems and Control, ETH Zurich, Zurich, Switzerland*

[2]*Bosch Center for Artificial Intelligence, Renningen, Germany*

Editors: A. Bayen, A. Jadbabaie, G. J. Pappas, P. Parrilo, B. Recht, C. Tomlin, M. Zeilinger

Abstract

Established techniques for simulation and prediction with Gaussian process (GP) dynamics often implicitly make use of an independence assumption on successive function evaluations of the dynamics model. This can result in significant error and underestimation of the prediction uncertainty, potentially leading to failures in safety-critical applications. This paper discusses methods that explicitly take the correlation of successive function evaluations into account. We first describe two sampling-based techniques; one approach provides samples of the true trajectory distribution, suitable for 'ground truth' simulations, while the other draws function samples from basis function approximations of the GP. Second, we propose a linearization-based technique that directly provides approximations of the trajectory distribution, taking correlations explicitly into account. We demonstrate the procedures in simple numerical examples, contrasting the results with established methods.

1. Introduction

Gaussian process (GP) regression has become a popular tool for learning dynamic systems from data, since it is flexible, requires little prior process knowledge and inherently provides a measure of model uncertainty by providing a probabilistic distribution over function values. Various use-cases have been proposed in the literature, such as state estimation (Ko and Fox (2008); Deisenroth et al. (2009a)), model-based reinforcement learning (Kuss and Rasmussen (2004); Deisenroth et al. (2009b, 2015)) and model predictive control (Klenske et al. (2013); Kocijan (2016); Ostafew et al. (2016); Kamthe and Deisenroth (2018); Hewing et al. (2019b); Kabzan et al. (2019)), see also Hewing et al. (2020).

All these problems require predictions of state x under input u for dynamic systems

$$x_{k+1} = f(x_k, u_k) + w_k, \tag{1}$$

where w_k are i.i.d. Gaussian disturbances and the uncertain dynamics function is distributed according to a Gaussian process $f \sim \mathcal{GP}$. The resulting stochastic state distributions over a prediction horizon typically need to be numerically approximated, since no closed-form solution exists. Established efforts are based on successive approximate and independent

evaluations of f from uncertain inputs at each time step, such that the approximate distribution of the state trajectory can be iteratively computed (Girard et al. (2003); Pan et al. (2017)). Such an approach implicitly introduces an independence assumption for successive evaluations of the function f, and neglects the fact that GPs describe a distribution over functions, such that successive evaluations of f are typically highly correlated. Figure 1 *(top)* shows illustrations of function samples from different GPs. Simulations with GP models avoiding this assumption have recently been discussed in Umlauft et al. (2018); Bradford et al. (2019) and are used for inference in GP state space models (Ialongo et al. (2019)). In many previous results, however, both theoretical analyses (e.g. in Vinogradska et al. (2016); Beckers and Hirche (2016); Polymenakos et al. (2019)) and simulations (e.g. in Girard et al. (2003)) are carried out under the independence assumption.

The goal of this paper is to provide simulation and prediction methods for system (1) which take correlations of successive GP function evaluations into account. Since the non-parametric stochastic description of the dynamics in (1) as a GP can be intuitively less accessible and challenging to simulate, we begin by providing an interpretation of the independence assumption in a motivating example using an analogy to parametric stochastic systems. The assumption then corresponds to considering all uncertainty as process noise, rather than constant uncertain model parameters. We address the correlation in successive function evaluations in (1) with methods for sampling-based simulation and for direct approximation of the predicted state trajectory distribution. We first discuss two sampling-based procedures: one provides trajectory samples over the prediction horizon from the true trajectory distribution, similarly proposed e.g. by Umlauft et al. (2018), for which we provide an alternative view in terms of the joint distributions of the trajectory, highlighting the specific computational structure and complexity. We contrast this ground-truth simulation against a second simulation approach based on approximate function samples, enabled by basis function GP approximation methods, similarly utilized in Bradford et al. (2018) in the context of Bayesian optimization. For direct approximation of the trajectory distribution, we then propose a modification of established linearization-based uncertainty propagation methods (Girard et al. (2003)), which takes the correlation of successive function evaluations into account. The procedures are demonstrated by revisiting the motivating example.

Appendix A in Hewing et al. (2019a) provides relevant background material on vector-valued GP regression and a more detailed definition of some of the notation used, which we believe to be intuitively accessible to readers familiar with GP regression.

2. Motivating Example

Since the accurate simulation of GP dynamics (1) is not straightforward (see Section 3.1), we begin by illustrating the implications of the independence assumption through simple parametric proxy systems, for which trajectory samples can be easily obtained. As we will show by revisiting the example in Section 4, these systems can be understood as limit cases of certain GP dynamics with either increasing or vanishing kernel length scale, illustrated in Figure 1 *(top)*. We consider scalar autonomous systems subject to i.i.d. noise $w_k \sim \mathcal{N}(0, 1)$ and either an uncertain element $\theta \sim \mathcal{N}(0, \sigma_f^2)$ which is constant for all time steps, or a changing $\theta_k \sim \mathcal{N}(0, \sigma_f^2)$ independently and identically distributed. The considered proxy systems are
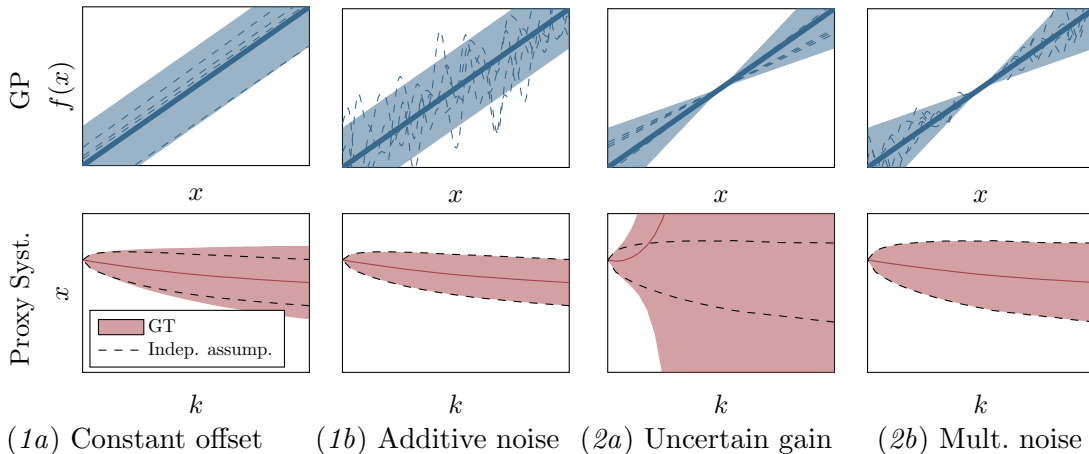
(*1a*) Constant offset   (*1b*) Additive noise   (*2a*) Uncertain gain   (*2b*) Mult. noise

Figure 1: *(top)*: Different GPs with mean, 2-σ variance and some function samples; the pairs *(1a)/(1b)* and *(2a)/(2b)* lead to the same mean and variance, respectively, when evaluated at any x independently. *(bottom)*: Mean and 2-σ variance of 20 000 simulated trajectories from proxy systems of the above GPs. Dashed lines illustrate the variance from simulations under the independence assumption, which leads to identical results for *(1a)* and *(1b)* as well as *(2a)* and *(2b)*.

(1a) Constant offset: $x_{k+1} = 0.95x_k + \theta + w_k$. This is a proxy for a GP prior with mean function $\mu(x) = 0.95x$ and squared exponential (SE) kernel $k_{\mathrm{SE}}(x,x') = \sigma_f^2 \exp(-\frac{(x-x')^2}{2\ell^2})$ with large length scale $\ell = 10$.

(1b) Additive noise: $x_{k+1} = 0.95x_k + \theta_k + w_k$. This is a proxy for a GP prior with mean function $\mu(x) = 0.95x$ and SE kernel with small length scale $\ell = 0.1$.

(2a) Uncertain gain: $x_{k+1} = (0.95 + \theta)x_k + w_k$. This is a proxy for a GP prior with mean function $\mu(x) = 0.95x$ and a linear kernel $k_{\mathrm{lin}}(x,x') = \sigma_f^2 xx'$.

(2b) Multiplicative noise: $x_{k+1} = (0.95 + \theta_k)x_k$. This is a proxy for a GP prior with mean function $\mu(x) = 0.95x$ and multiplied linear/SE kernel with small length scale $\ell = 0.1$.

The plots in Figure 1 *(bottom)* show mean and variance information from 20 000 simulated state trajectories of these proxy systems. The independent uncertainty θ_k at each time step in *(1b)* and *(2b)*, in contrast to random but constant θ in *(1a)* and *(2a)*, leads to a significantly smaller spread of system trajectories, i.e. significantly smaller 'predicted' uncertainty as indicated by the variance in Figure 1. Often times, methods for prediction and simulation with GP dynamics consider successive evaluations of the dynamics f as independent. This corresponds to drawing the persistent uncertain parameter θ in *(1a)* and *(2a)* independently at each time step, i.e. identical to θ_k in *(1b)* and *(2b)*, which leads to a significant underestimation of the resulting uncertainty, shown with dashed lines in Figure 1 *(bottom)*.

3. Predictions with GP Dynamics

In this paper, we provide simulation and prediction methods for system (1) which take correlations of successive GP function evaluations into account and enable accurate predictions in examples *(1a)* and *(2a)*. For simplicity, we focus on autonomous systems

$$x_{k+1} = f(x_k) + w_k, \ f \sim \mathcal{GP}(\mu, k), \tag{2}$$

subject to i.i.d. noise $w_k \sim \mathcal{N}(0, Q)$. The distribution of f is specified by a GP with mean function $\mu : \mathbb{R}^n \to \mathbb{R}^n$, which in vector-valued regression maps to a vector, and the kernel $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n \times n}$, which maps to a positive (semi-)definite variance matrix. All methods can be applied similarly to controlled systems, see Appendix C in Hewing et al. (2019a).

In the following, we use respective capital letters for stacked vectors, e.g. $X = [x_1^\mathsf{T}, \ldots, x_N^\mathsf{T}]^\mathsf{T}$ and $X_{a:b}$ with indices $a < b$ to refer to $[x_a^\mathsf{T}, \ldots, x_b^\mathsf{T}]^\mathsf{T}$. With a slight abuse of notation we overload μ and k when evaluated on stacked vectors, such that $\mu(X)_i = \mu(x_i)$ and $k(X, X')_{i,j} = k(x_i, x_j')$. With this, we write evaluations of f at $x_a, \ldots, x_b$ compactly as $[f_a^\mathsf{T}, \ldots, f_b^\mathsf{T}]^\mathsf{T} \sim \mathcal{N}(\mu(X_{a:b}), k(X_{a:b}, X_{a:b}'))$.

Remark 1 (Inference) *Inference given data $\mathcal{D} = \{X^t, Y^t\}$ is carried out by conditioning the Gaussian distributions on the collected data points. This can be expressed as modified mean and kernel function*

$$\mu^\mathcal{D}(x) = \mu(x) + k(x, X^t)(k(X^t, X^t) + I \otimes Q)^{-1}(Y^t - \mu(X^t)),$$
$$k^\mathcal{D}(x, x') = k(x, x') - k(x, X^t)(k(X^t, X^t) + I \otimes Q)^{-1}k(X^t, x').$$

To simplify notation, we therefore refer simply to μ and k for the remainder of the paper, which can similarly denote a GP conditioned on data.

Under the GP assumption, all evaluations of f are jointly Gaussian distributed according to the specified mean and kernel function. Using (2), we can therefore express the distribution of the predicted state trajectory over N time steps implicitly as

$$\begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu(x_0) \\ \vdots \\ \mu(x_{N-1}) \end{bmatrix}, \begin{bmatrix} k(x_0, x_0) + Q & \ldots & k(x_0, x_{N-1}) \\ \vdots & \ddots & \vdots \\ k(x_{N-1}, x_0) & \ldots & k(x_{N-1}, x_{N-1}) + Q \end{bmatrix} \right), \tag{3}$$

which, using shorthand notation, can be compactly expressed as

$$X_{1:N} \sim \mathcal{N}(\mu(X_{0:N-1}), k(X_{0:N-1}, X_{0:N-1}) + I \otimes Q).$$

This implicit description of the predicted trajectory is a challenging object to deal with, since the (shifted) state sequence appears on both the left- and right-hand side.

Remark 2 (Independence Assumption) *The common independence assumption can be understood as approximating (3) using a block diagonal covariance matrix*

$$\begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu(x_0) \\ \vdots \\ \mu(x_{N-1}) \end{bmatrix}, \begin{bmatrix} k(x_0, x_0) + Q & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & k(x_{N-1}, x_{N-1}) + Q \end{bmatrix} \right).$$

In the following, we provide sampling-based methodologies that avoid the independence assumption, allowing for accurate simulation of the dynamic system, as well as a linearization-based direct approximation of the trajectory distribution (3).

3.1. Sampling-Based Simulation with GP Dynamics

Sampling-based simulation of systems given by GP dynamics is of great interest, e.g. for particle filtering or simulation-based or -aided controller and system design. A naive approach is to draw N_s *function samples* $f^{(i)} \sim \mathcal{GP}(\mu, k)$, $i = 1, \ldots, N_s$, as shown in Figure 1. These function samples can then be used in simulation to generate state trajectory samples $X^{(i)} = [(x_0^{(i)})^\mathsf{T}, \ldots, (x_N^{(i)})^\mathsf{T}]^\mathsf{T}$. Generally, function samples can only be obtained approximately, which is typically done by evaluating f on a fine grid spanning the entire domain and subsequent sampling from the resulting joint Gaussian distribution. It therefore comes with the usual limitations associated with gridding, such as very poor scalability to systems of higher dimensions.

Here, we discuss two sampling-based approaches alleviating this drawback. The first approach directly generates samples of the *trajectory* from the true trajectory distribution, circumventing the need to draw a function sample. The second approach generates *approximate* function samples based on basis function approximations of the GP. This avoids gridding by providing explicit representations of the sampled functions and can have computational advantages over the direct trajectory sampling.

3.1.1. TRAJECTORY SAMPLING

Given the initial state x_0, we generate trajectory samples $X^{(i)}$ which are consistent with the GP (3) by drawing samples of subsequent time steps, starting with the first step

$$x_1^{(i)} = \mu(x_0) + \sqrt{k(x_0, x_0) + Q} \, \tilde{w}_0^{(i)},$$

where $\sqrt{\cdot}$ denotes the Cholesky decomposition and $\tilde{w}_0^{(i)} \sim \mathcal{N}(0, I)$ is drawn from the standard normal distribution. Samples of the following state $x_2^{(i)}$ can then be drawn by conditioning the GP on the respective realization, i.e. $\mathcal{D}_1^{(i)} = \{x_1^{(i)}, x_0^{(i)}\}$ in Remark 1, see also Umlauft et al. (2018). Equivalently, one can consider the joint distribution of $x_1^{(i)}$ and $x_2^{(i)}$, avoiding the computation of the conditional distributions and revealing the particular structure of the problem

$$\begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \end{bmatrix} = \begin{bmatrix} \mu(x_0) \\ \mu(x_1^{(i)}) \end{bmatrix} + \sqrt{\begin{bmatrix} k(x_0, x_0) + Q & k(x_0, x_1^{(i)}) \\ k(x_1^{(i)}, x_0) & k(x_1^{(i)}, x_1^{(i)}) + Q \end{bmatrix}} \begin{bmatrix} w_0^{(i)} \\ w_1^{(i)} \end{bmatrix}.$$

Given the sample $x_1^{(i)}$ of the first time step, the Cholesky decomposition for the joint distribution can be computed, and $x_2^{(i)}$ generated using samples of the standard normal distribution $w_1^{(i)} \sim \mathcal{N}(0, I)$. Iterating this procedure, a sample up to time step $k + 1$ is generated by

$$X_{1:k+1}^{(i)} = \mu(X_{0:k}^{(i)}) + \sqrt{k(X_{0:k}^{(i)}, X_{0:k}^{(i)}) + I \otimes Q} \, \tilde{W}_{0:k+1}^{(i)}, \tag{4}$$

where $X^{(i)}$ is the trajectory sample, and $\tilde{W}^{(i)} = [(\tilde{w}_0^{(i)})^\mathsf{T}, \ldots, (\tilde{w}_{N-1}^{(i)})^\mathsf{T}]^\mathsf{T}$ the corresponding sample drawn from the standard normal distribution. Since there is no approximation involved, this procedure can be used for *ground truth* simulations and serves as a basis for

comparison against other methods in the following. Note that it is not necessary to recalculate the entire Cholesky decomposition at each time step, but only the added lines of each time step, cf. the standard algorithm for Cholesky decomposition (Demmel, 1997, Ch. 2.7). The computational complexity is therefore related to *one* full Cholesky decomposition and scales cubically with the prediction horizon.

3.1.2. APPROXIMATE FUNCTION SAMPLES BASED ON BASIS FUNCTIONS

The computational cost of drawing trajectory samples from the true distribution grows rapidly with the prediction horizon, while generating function samples based on gridding scales poorly with the state dimension n. This motivates an alternative procedure of generating explicit representations of *approximate* function samples to be used for simulation–resulting in computational complexity that is linear in the prediction length while scaling to higher dimensional systems. The approach is based on the fact that GPs can be written as a linear combination of a (possibly infinite) number of basis functions

$$f(x) = \phi(x)^{\mathsf{T}}\theta, \text{ with } \theta \sim \mathcal{N}(\mu_\theta, \Sigma_\theta),$$

where the kernel is given by $k(x, x') = \phi(x)^{\mathsf{T}}\phi(x')$ and $\mu_\theta, \Sigma_\theta$ follow from Bayesian linear regression, see e.g. Rasmussen and Williams (2006, Ch. 2.1). The function f can then be approximated by a *finite* subset of m basis functions, and an explicit approximate function sample $\tilde{f}^{(i)} = \sum_{j=1}^{m} \phi_j(x)\theta_j^{(i)}$ can be constructed by sampling the weight vector θ.

In general, computing basis functions $\phi(x)$ for a particular kernel can be challenging. A common finite dimensional approximation of ϕ is given by the first m kernel eigenfunctions, which can be approximated by the Nyström method if the solutions are not analytically tractable (Press et al., 2007, Ch. 19.1). For continuous shift-invariant kernels, Rahimi and Recht (2008) have shown that cosine functions with random frequencies and phase shifts are particularly well-suited as basis functions, leading to the so-called sparse spectrum GP approximation (Lázaro-Gredilla et al., 2010), the use of which was proposed in Bradford et al. (2018) to draw approximate function samples in the context of Bayesian optimization. Typically, the kernel approximations recover the exact distribution of f as the number of basis functions approaches infinity. Appendix B in Hewing et al. (2019a) includes a short discussion of common basis function approximations as used in our numerical examples in Section 4.

3.2. Linearization-Based Approximation

In many applications, such as predictive control, the computational cost of sampling-based GP predictions can be prohibitive, motivating efficient direct approximations of the predictive trajectory distributions. In the following, we propose a linearization-based approximation method which can be understood as a modification of established prediction techniques. The procedure is based on approximating the distribution of the predicted state sequence by iteratively linearizing the mean function μ around the previous state mean. We follow a similar procedure as in Section 3.1.1 to obtain

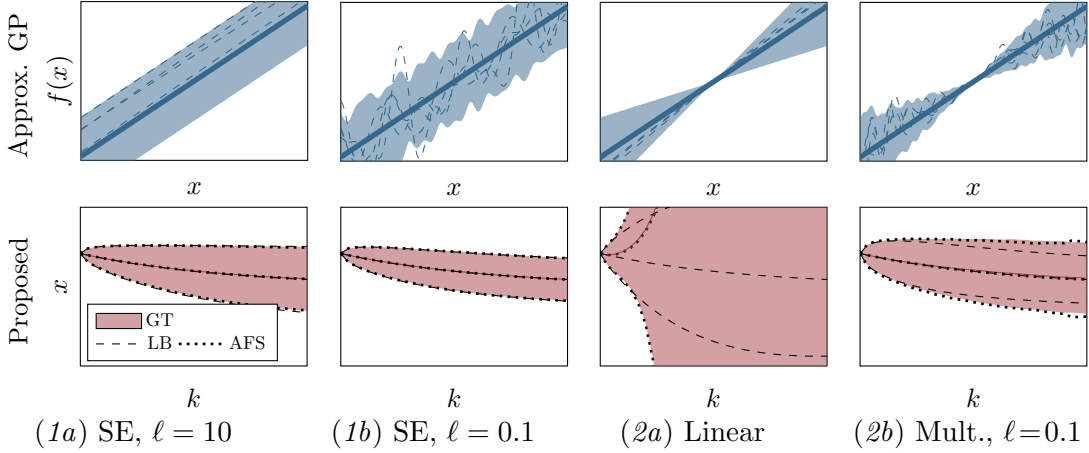$$x_1 = \mu(x_0) + \sqrt{k(x_0, x_0) + Q}\,\tilde{w}_0$$

Figure 2: *(top):* Approximate GPs using 10 basis functions. *(bottom):* Proposed prediction techniques. Mean and 2-σ variance of 20 000 ground truth (GT) simulations using trajectory sampling in red. The results using approximate function sampling (AFS) as dotted lines and linearization-based (LB) prediction as dashed lines.

in which again $\tilde{w}_0 \sim \mathcal{N}(0, I)$. Similar to Girard et al. (2003), we consider a first-order Taylor expansion of μ around $\mu_1 = \mu(x_0)$ for the next time step, but consider the zero-order expansion of the *joint* variance k. This leads to

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \approx \begin{bmatrix} \mu_1 \\ \mu(\mu_1) \end{bmatrix} + \begin{bmatrix} I & 0 \\ \nabla\mu(\mu_1) & I \end{bmatrix} \sqrt{\begin{bmatrix} k(x_0, x_0) + Q & k(x_0, \mu_1) \\ k(\mu_1, x_0) & k(\mu_1, \mu_1) + Q \end{bmatrix}} \begin{bmatrix} \tilde{w}_0 \\ \tilde{w}_1 \end{bmatrix}.$$

Iterating this procedure and introducing the short-hand notation $\tilde{W} = [\tilde{w}_0^\mathsf{T}, \ldots, \tilde{w}_{N-1}^\mathsf{T}]^\mathsf{T}$ and $M = [\mu_0^\mathsf{T}, \ldots, \mu_N^\mathsf{T}]^\mathsf{T}$ in which $\mu_0 = x_0$ and $\mu_{k+1} = \mu(\mu_k)$, yields the expression

$$X_{1:N} \approx M_{1:N} + A\sqrt{k(M_{0:N-1}, M_{0:N-1}) + I \otimes Q}\,\tilde{W},$$

in which A is a lower triangular block matrix with the relevant derivatives $A_{i,j} = \prod_{l=j}^{i-1} \nabla\mu(\mu_l)$. This results in the following approximate distribution of the state trajectory

$$X_{1:N} \dot\sim \mathcal{N}(M_{1:N}, A(k(M_{0:N-1}, M_{0:N-1}) + I \otimes Q)A^\mathsf{T}). \tag{5}$$

In contrast to established prediction methods using the independence assumption, we approximate the full variance matrix in (3) instead of assuming a block diagonal structure, see Remark 2. The proposed technique therefore corrects for correlation of subsequent function evaluations along the mean prediction. Note that no Cholesky decomposition is necessary for (5) or computation of the mean trajectory $M_{1:N}$. The computational complexity therefore scales quadratically with the prediction horizon, as opposed to a linear scaling under the independence assumption with a block diagonal structure.

4. Example Revisited

In order to demonstrate the proposed simulation and prediction methods, we revisit the motivating example of Section 2. We make use of these simple examples, i.e. GPs without explicitly considering data, to clearly illustrate the effects of different techniques and assumptions. The techniques developed, however, apply similarly to GPs conditioned on data (cf. Remark 1) or controlled systems (see Appendix C in Hewing et al. (2019a)). The trajectory sampling technique described in Section 3.1.1 allows us to simulate the actual GP dynamics described in Section 2 and shown in Figure 1 *(top)*, instead of the proxy systems.

We compare *ground truth* simulations using trajectory sampling (GT) to simulations using approximate function samples (AFS) and the developed linearization-based prediction technique (LB). Approximate function samples are drawn using 10 random basis functions of the SE kernels for examples *(1a)* and *(1b)*, which are transformed with a linear gain for *(2b)*. For *(2a)* we use the exact basis function representation by the single linear eigenfunction. Additional information can be found in Appendix B in Hewing et al. (2019a). The resulting basis function approximations are shown in Figure 2 *(top)*. Note that the approximation with as little as 10 basis functions is possible due to the simple structure of the example. For complex applications, good approximations can typically be found using hundreds to a few thousand basis functions (Lázaro-Gredilla et al., 2010; Hernández-Lobato et al., 2014).

The results with the techniques considered are shown in Figure 2 *(bottom)*. Ground truth simulations using the trajectory sampling in Section 3.1.1 show very good correspondence to the proxy systems investigated in Section 2 (cf. Figure 1). All the methods show a clear adjustment to the persistent uncertainties in *(1a)* and *(2a)*. This is in contrast to established prediction methods, which consider *(1a)* and *(2a)* identical to *(1b)* and *(2b)*, since the marginal distributions for a *single* evaluation are identical, as illustrated by the shaded regions in Figure 1 *(top)*. For examples *(1a)* and *(1b)*, the linearization-based uncertainty propagation technique developed in Section 3.2 is almost perfect. This is different in *(2a)* and *(2b)*, where the linearization-based technique leads to significant errors due to the local approximation around the mean. Note here in particular, that as the predicted mean approaches the origin, the local approximation of k approaches zero. Simulation based on basis function approximations yields almost indistinguishable results from the ground truth simulations. This suggests that good results can be obtained at significantly lower computational cost than ground truth simulations based on approximate function samples, if a suitable basis function representation can be found.

5. Conclusion

We have shown that correlation between successive function evaluations in a GP dynamics system significantly influences the resulting state trajectory distributions. Two sampling-based methods and a proposed direct approximation of the trajectory distributions were compared and shown to take these correlations into account. The methods significantly improve prediction accuracy over established techniques based on an independence assumption, at the cost of increased computational demand for long horizons. We showed that approximate function samples obtained from basis function approximations of the GP can alleviate this drawback, allowing computationally efficient sampling-based simulations.

Acknowledgments

References

Thomas Beckers and Sandra Hirche. Equilibrium distributions and stability analysis of Gaussian process state space models. In *Conference on Decision and Control (CDC)*, pages 6355–6361, 2016.

Eric Bradford, Artur M. Schweidtmann, and Alexei Lapkin. Efficient multiobjective optimization employing Gaussian processes, spectral sampling and a genetic algorithm. *Journal of global optimization*, 71(2):407–438, 2018.

Eric Bradford, Lars Imsland, and Ehecatl Antonio del Rio-Chanona. Nonlinear model predictive control with explicit back-offs for Gaussian process state space models. In *Conference on decision and control (CDC)*, 2019.

Marc P. Deisenroth, Marco F. Huber, and Uwe D. Hanebeck. Analytic moment-based Gaussian process filtering. In *26th Annual International Conference on Machine Learning (ICML)*, pages 225–232, 2009a.

Marc P. Deisenroth, Carl E. Rasmussen, and Jan Peters. Gaussian process dynamic programming. *Neurocomputing*, 72(7):1508 – 1524, 2009b.

Marc P. Deisenroth, Dieter Fox, and Carl E. Rasmussen. Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):408–423, 2015.

James W. Demmel. *Applied numerical linear algebra*. SIAM, 1997.

Agathe Girard, Carl E. Rasmussen, Joaquin Quiñonero Candela, and Roderick Murray-Smith. Gaussian process priors with uncertain inputs - application to multiple-step ahead time series forecasting. In *Advances in Neural Information Processing Systems 15*, pages 545–552, 2003.

José Miguel Hernández-Lobato, Matthew W. Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in Neural Information Processing Systems 27*, pages 918–926. 2014.

Lukas Hewing, Elena Arcari, Lukas P. Fröhlich, and Melanie N. Zeilinger. On simulation and trajectory prediction with Gaussian process dynamics. *arXiv preprint*, 2019a.

Lukas Hewing, Juraj Kabzan, and Melanie N. Zeilinger. Cautious model predictive control using Gaussian process regression. *IEEE Transactions on Control Systems Technology*, 2019b. *(in press)*.

Lukas Hewing, Kim P. Wabersich, Marcel Menner, and Melanie N. Zeilinger. Learning-based model predictive control: Toward safe learning in control. *Annual Review of Control, Robotics, and Autonomous Systems*, Vol. 3, 2020. *(in press)*.

Alessandro Davide Ialongo, Mark Van Der Wilk, James Hensman, and Carl Edward Rasmussen. Overcoming mean-field approximations in recurrent Gaussian process models. In *International Conference on Machine Learning*, pages 2931–2940, 2019.

Juraj Kabzan, Lukas Hewing, Alexander Liniger, and Melanie N. Zeilinger. Learning-based model predictive control for autonomous racing. *IEEE Robotics and Automation Letters*, 4(4):3363–3370, 2019.

Sanket Kamthe and Marc P. Deisenroth. Data-efficient reinforcement learning with probabilistic model predictive control. In *21st International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2018.

Edgar D. Klenske, Melanie N. Zeilinger, Bernhard Schölkopf, and Philipp Hennig. Nonparametric dynamics estimation for time periodic systems. In *51st Annual Allerton Conference on Communication, Control, and Computing*, pages 486–493, 2013.

Jonathan Ko and Dieter Fox. GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3471–3476, 2008.

Juš Kocijan. *Modelling and control of dynamic systems using Gaussian process models*. Springer International Publishing, 2016.

Malte Kuss and Carl E. Rasmussen. Gaussian processes in reinforcement learning. In *Advances in Neural Information Processing Systems 16*, pages 751–758. 2004.

Miguel Lázaro-Gredilla, Joaquin Quiñonero-Candela, Carl E. Rasmussen, and Aníbal R. Figueiras-Vidal. Sparse spectrum Gaussian process regression. *Journal of Machine Learning Research*, 11:1865–1881, 2010.

Chris J. Ostafew, Angela P. Schoellig, and Timothy D. Barfoot. Robust constrained learning-based NMPC enabling reliable mobile robot path tracking. *International Journal of Robotics Research*, 35(13):1547–1563, 2016.

Yunpeng Pan, Xinyan Yan, Evangelos A. Theodorou, and Byron Boots. Prediction under uncertainty in sparse spectrum Gaussian processes with applications to filtering and control. In *34th International Conference on Machine Learning*, pages 2760–2768, 2017.

Kyriakos Polymenakos, Luca Laurenti, Andrea Patane, Jan-Peter Calliess, Luca Cardelli, Marta Kwiatkowska, Alessandro Abate, and Stephen Roberts. Safety guarantees for planning based on iterative Gaussian processes. *arXiv:1912.00071*, 2019.

William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes*. Cambridge University Press, 2007.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems 20*, pages 1177–1184, 2008.

Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

Jonas Umlauft, Thomas Beckers, and Sandra Hirche. Scenario-based optimal control for Gaussian process state space models. In *2018 European Control Conference (ECC)*, pages 1386–1392, 2018.

Julia Vinogradska, Bastian Bischoff, Duy Nguyen-Tuong, Anne Romer, Henner Schmidt, and Jan Peters. Stability of controllers for Gaussian process forward models. In *International Conference on Machine Learning*, pages 545–554, 2016.