Confidence Regions for Simulations with Learned Probabilistic Models*

Armin Lederer[1], Qing Hao, Sandra Hirche[1]

Abstract— **Due to the growing amount of data and processing capabilities, machine learning techniques are increasingly applied for the identification of dynamical systems. Especially probabilistic methods are promising for learning models, which in turn are frequently used for simulations. Although confidence regions around predicted trajectories are of crucial importance in many control approaches, few rigorous mathematical analysis methods are available for learned probabilistic models. Therefore, we propose a novel method to estimate confidence regions for predicted trajectories, and assign them a confidence level based on Monte Carlo random trajectory sampling. Since the confidence level has a strongly nonlinear dependence on the number of Monte Carlo samples, we derive a lower bound on the number of samples that ensures a desired minimum confidence level. The efficiency and flexibility of the proposed method is demonstrated in simulations of a Bayesian hidden Markov model and a Gaussian process state space model.**

I. Introduction

With the ever growing amount of available data and continuously improving data processing capabilities, supervised machine learning is becoming more and more popular in system identification for control design and predicting system behavior. Especially when the lack of knowledge about the system structure prevents the application of classical parametric system identification methods, modern regression techniques, such as Gaussian processes and deep learning, provide a flexible and efficient alternative. Probabilistic machine learning methods [1] have been gaining increasing attention, as they provide a measure of model fidelity, which considers uncertainty sources such as noisy training data and unknown model structures. Due to this beneficial property, these methods have found many applications in control, ranging from feedback linearization [2] to reinforcement learning [3]. Especially in simulation-based (or prediction-based) approaches, information about the model fidelity can be included advantageously into control. For example, the consideration of the uncertainty of learned models leads to cautiousness in model predictive control [4], and can be efficiently implemented using scenario approaches [5]. Furthermore, uncertainty awareness in model-based reinforcement learning automatically balances system identification and task execution [6], and allows to anticipate the effect of online learning during policy optimization [7].

Despite of this prevalence of learned probabilistic models in simulation-based control approaches, the uncertainty along the predicted trajectories is usually approximated without any theoretical guarantees. While this is not a problem when considering the uncertainty in cost functions, the lack of guarantees prevents the straightforward computation of confidence regions around predicted trajectories, which are required to enforce constraints on the real system. To the best of our knowledge, statistically sound confidence regions for simulations with learned probabilistic models have not been analyzed in general. One exception is [8], where a multiple-step prediction error bound for Gaussian process regression is derived. However, it is based on a uniform error bound and deterministic uncertainty propagation, which leads to conservative confidence regions, growing almost exponentially over the prediction horizon. Therefore, this approach is not suitable for long horizons typically used in reinforcement learning or model predictive control.

The contribution of this paper is a novel method for determining confidence regions around trajectories predicted with models obtained by probabilistic machine learning techniques. We propose a Monte Carlo method to sample from generally unknown trajectory distributions, and prove that the confidence of a set can be calculated based on the sampled random trajectories. In order to facilitate the application of our result, we derive a lower bound for the required number of sample trajectories, such that the desired confidence level is achieved. The flexibility and tightness compared to existing methods is demonstrated in numerical simulations.

The remainder of this paper is structured as follows: After the formal problem statement in Section II, we introduce the basic Monte Carlo algorithm in Section III. Section IV proposes the trajectory sampling approach and derives confidence levels based on random samples. The confidence sets and corresponding levels are evaluated for hidden Markov models and Gaussian process models, in Section V.

II. Problem Formulation

Consider a dynamical system[1]

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k) + \boldsymbol{\epsilon}_k \tag{1a}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_{k+1}) + \boldsymbol{\eta}_k \tag{1b}$$

with state $\boldsymbol{x}_k \in \mathbb{X}$, initial state $\boldsymbol{x}_0 \in \mathbb{X}$, observation $\boldsymbol{y}_k \in \mathbb{Y}$, unknown functions $\boldsymbol{f} : \mathbb{X} \to \mathbb{X}$, $\boldsymbol{h} : \mathbb{X} \to \mathbb{Y}$, i.i.d. process noise $\boldsymbol{\epsilon}_k \in \mathbb{X}$ defined on a probability space $(\mathbb{X}, \mathcal{F}_\epsilon, \mathbb{P}_\epsilon)$

*A. L. gratefully acknowledges financial support from the German Academic Scholarship Foundation 10.23919/ACC45564.2020.9147978.

[1]Armin Lederer and Sandra Hirche are with the Chair of Information-oriented Control (ITR), Department of Electrical and Computer Engineering, Technical University of Munich, Germany {armin.lederer,hirche}@tum.de

[1]Lower/upper case bold symbols denote vectors/matrices, $\mathbb{R}_+$ all non-negative real numbers, $\mathbb{N}$ all integers, $\mathbb{E}[\cdot]$ the expected value of a random variable, $\mathbb{P}(\cdot)$ a probability measure, and other blackboard bold symbols subsets of $\mathbb{R}$ and $\mathbb{N}$, respectively. The Gaussian distribution with mean μ and standard deviation σ is denoted by $\mathcal{N}(\mu, \sigma)$ and random samples are symbolized by a superscript (n).

with σ-algebra $\mathcal{F}_\epsilon$ and probability distribution $\mathbb{P}_\epsilon$, and i.i.d. observation noise $\boldsymbol{\eta}_k \in \mathbb{Y}$ with σ-algebra $\mathcal{F}_\eta$ and probability distribution $\mathbb{P}_\eta$. Depending on the considered problem, the state and output spaces can be continuous, e.g., $\mathbb{X}, \mathbb{Y} \subset \mathbb{R}^d$, or discrete, e.g., $\mathbb{X}, \mathbb{Y} \subset \mathbb{N}$. Based on measurements of the dynamical system, probabilistic machine learning methods can be applied to obtain a stochastic model in the form of random fields $\{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})\}_{\boldsymbol{x}\in\mathbb{X}}$ and $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x}\in\mathbb{X}}$. These models are defined on probability spaces $(\Omega, \mathcal{F}_\omega, \mathbb{P}_\omega)$ and $(\Psi, \mathcal{F}_\psi, \mathbb{P}_\psi)$, such that the random variables $\boldsymbol{\omega}$ and $\boldsymbol{\psi}$ can be considered as (potentially infinite-dimensional) vectors of uncertain model parameters. In order to be able to provide guarantees for predictions with these stochastic models, it is necessary that the stochastic models are well calibrated. This is formalized in the following assumption.

Assumption 1: The unknown functions $\boldsymbol{f}(\cdot)$ and $\boldsymbol{h}(\cdot)$ are realizations of the stochastic model $\{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})\}_{\boldsymbol{x}\in\mathbb{X}}$ and $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x}\in\mathbb{X}}$.

While it can be difficult to ensure a good calibration in practice, bounds on the misspecification of stochastic models have been derived for several probabilistic machine learning methods, see, e.g., [9]. Therefore, possible model errors can easily be integrated in any subsequent analysis if Assumption 1 is violated.

We want to use the stochastic models $\{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})\}_{\boldsymbol{x}\in\mathbb{X}}$ and $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x}\in\mathbb{X}}$ to predict the behavior of the state and output trajectories $\boldsymbol{x}_k, \boldsymbol{y}_k$, $k = 0, \dots, K$ of the unknown real world system (1) up to some horizon $K \in \mathbb{N}$. The computation of the distributions $p(\boldsymbol{x}_k)$, $p(\boldsymbol{y}_k)$ can be computationally demanding for large discrete state models and is generally intractable for continuous state random dynamical systems. Therefore, we have to approximate the exact distributions leading to predictive distributions $\hat{p}(\boldsymbol{x}_k)$, $\hat{p}(\boldsymbol{y}_k)$, which can usually provide only limited information about the true predictive distribution, see, e.g., [10]. However, in safety critical applications, it is important to obtain reliable confidence levels $\delta \in \mathbb{R}_+$ of given confidence regions $\mathbb{S}_k \subset \mathbb{X}$, $\mathbb{T}_k \subset \mathbb{Y}$, $k = 0, \dots, K$. Hence, our goal is the development of a method to determine the minimum value of δ, such that

$$\mathbb{P}\left(\boldsymbol{x}_k \in \mathbb{S}_k, \boldsymbol{y}_k \in \mathbb{T}_k, \quad \forall k = 0, \dots, K\right) \geq 1 - \delta \tag{2}$$

is guaranteed under the joint probability measure $\mathbb{P}$.

In order to achieve this, knowledge of the involved distributions is required. We distinguish two assumptions, which are applied in a mutually exclusive fashion in the sequel.

Assumption 2: We can draw random samples from the probability distributions $\mathbb{P}_\omega$, $\mathbb{P}_\psi$, $\mathbb{P}_\epsilon$ and $\mathbb{P}_\eta$.

Assumption 3: The probability measures $\mathbb{P}_\epsilon$ and $\mathbb{P}_\eta$ are known and we can draw samples from the corresponding distributions. Furthermore, we can draw random samples from the Bayesian updates $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}}$ and $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_k}|\boldsymbol{H}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{H}_{\boldsymbol{x}_0}}$ of the random variables $\boldsymbol{F}_{\boldsymbol{x}_k} := \boldsymbol{F}(\boldsymbol{x}_k, \boldsymbol{\omega})$ and $\boldsymbol{H}_{\boldsymbol{x}_k} := \boldsymbol{H}(\boldsymbol{x}_k, \boldsymbol{\psi})$ for all $k = 0, \dots, K$.

Assumption 2 is applied in case we have a model parameterized by a finite number of random variables, as outlined in Sec. IV-A. This is a suitable abstraction of many models obtained from statistical machine learning, e.g., probabilistic neural networks [11] and Bayesian hidden Markov models [12]. Even when no analytical expression for the distribution of the random variables $\boldsymbol{\omega}$, $\boldsymbol{\psi}$ can be calculated, samples can often be drawn by applying Markov Chain Monte Carlo methods [13]. However, Assumption 2 is not suited for non-parametric regression methods, such as Gaussian process regression, because the resulting models depend on infinite-dimensional random vectors [14]. In this case, we apply Assumption 3, which allows to iteratively update the conditional distributions $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}}$ and $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_k}|\boldsymbol{H}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{H}_{\boldsymbol{x}_0}}$, and draw the next function values. This approach is described in Sec. IV-B.

III. Numerical Integration with Monte Carlo Sampling

Determining expectations and probabilities, such as those arising in multiple-step predictions, typically requires the computation of intractable integrals. Monte Carlo sampling can be used to approximate integrals of the form

$$E = \int_{\mathbb{A}} g(\boldsymbol{x}) p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{3}$$

where $p : \mathbb{X} \to \mathbb{R}_+$ is a predetermined probability density, $g : \mathbb{X} \to \mathbb{R}$ is an arbitrary function and $\mathbb{A} \subset \mathbb{X}$ is a predetermined set. The idea behind Monte Carlo sampling is to generate N i.i.d. random samples $\boldsymbol{x}^{(n)}$, which follow the probability distribution corresponding to the density $p(\cdot)$. Then, the exact integral is approximated by its empirical value [15]

$$\hat{E} = \frac{1}{N} \sum_{n=1}^{N} g\left(\boldsymbol{x}^{(n)}\right). \tag{4}$$

The strength of this approximation is its statistical foundation. Due to the strong law of large numbers, $\hat{E}$ is guaranteed to converge to the unknown integral almost surely in the limit of infinitely many random samples, i.e.,

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} g\left(\boldsymbol{x}^{(n)}\right) = E \quad \text{a.s.} \tag{5}$$

Furthermore, the variance of this approximation is typically used as a first order error estimate [16], which has been shown to converge to zero in the order of $\frac{1}{\sqrt{N}}$. Therefore, the convergence rate is independent of the state space dimension. When approximating probabilities with (4), the indicator function is employed [15]:

Definition 1: The indicator function $I_{\mathbb{A}} : \mathbb{X} \to \{0, 1\}$ for a set $\mathbb{A} \subset \mathbb{X}$ is defined as

$$I_{\mathbb{A}}(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \in \mathbb{A} \\ 0 & \text{if } \boldsymbol{x} \notin \mathbb{A}. \end{cases} \tag{6}$$

Using this definition, the empirical probability of a set $\mathbb{A}$ can be seen as a special case of (4) with $g(\cdot) = I_{\mathbb{A}}(\cdot)$.

IV. Confidence Regions for Trajectories of Learned Probabilistic Models

When a model of an unknown dynamical system is known, it is possible to approximate the real system be-

Algorithm 1: Sampling from parameter distributions

```
Function sample_P($N$, $\mathbb{P}_\epsilon$, $\mathbb{P}_\eta$, $\mathbb{P}_\omega$, $\mathbb{P}_\psi$, $\mathbb{P}_{\boldsymbol{x}_0}$):
    for $n = 1, \dots, N$ do
        Sample $\boldsymbol{\omega}^{(n)}$ and $\boldsymbol{\psi}^{(n)}$ from $\mathbb{P}_\omega$ and $\mathbb{P}_\psi$
        Sample $\boldsymbol{x}_0^{(n)}$ from $\mathbb{P}_{\boldsymbol{x}_0}$
        for $k = 0, \dots, K-1$ do
            Sample $\boldsymbol{\epsilon}_k^{(n)}$ and $\boldsymbol{\eta}_k^{(n)}$ from $\mathbb{P}_\epsilon$ and $\mathbb{P}_\eta$
            $\boldsymbol{Y}_{[k]}^{(n)} \leftarrow \boldsymbol{H}(\boldsymbol{x}_{k+1}^{(n)}, \boldsymbol{\omega}^{(n)}) + \boldsymbol{\eta}_k^{(n)}$
            $\boldsymbol{x}_{k+1}^{(n)} \leftarrow \boldsymbol{F}(\boldsymbol{x}_k^{(n)}, \boldsymbol{\psi}^{(n)}) + \boldsymbol{\epsilon}_k^{(n)}$
        Sample $\boldsymbol{\eta}_K^{(n)}$ from $\mathbb{P}_\eta$
        $\boldsymbol{Y}_{[K]}^{(n)} \leftarrow \boldsymbol{H}(\boldsymbol{x}_K^{(n)}, \boldsymbol{\psi}^{(n)}) + \boldsymbol{\eta}_K^{(n)}$
```

havior numerically by simulating sample trajectories of the stochastic model. We present the trajectory sampling approach for known parameter distributions in Section IV-A, while Section IV-B introduces our method for iteratively sampling from the conditional distributions $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}}$ and $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_k}|\boldsymbol{H}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{H}_{\boldsymbol{x}_0}}$. Finally, we derive confidence levels for given confidence regions in Section IV-C.

A. Trajectory Sampling from Parameter Distributions

Since we consider an unknown deterministic system $\boldsymbol{f}(\cdot)$, $\boldsymbol{h}(\cdot)$ perturbed by random noise $\boldsymbol{\epsilon}, \boldsymbol{\eta}$, a simulation of this system based on the random field models $\{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})\}_{\boldsymbol{x} \in \mathbb{X}}$, $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x} \in \mathbb{X}}$ should consider this structure. In order to describe the procedure for a suitable simulation of the model, we formally introduce the concept of state trajectories, which can be directly extended to output trajectories.

Definition 2: A trajectory $\boldsymbol{X}_{[0:K]}$ is defined as a sequence of states $\boldsymbol{x}_k$ resulting from iterative evaluation of the dynamical system (1) such that $\boldsymbol{X}_{[k]} = \boldsymbol{x}_k$, for all $k = 0, \dots, K$.

This definition implies that a trajectory $\boldsymbol{X}_{[0:K]}$ is generated by deterministic functions $f(\cdot)$, $h(\cdot)$, which are driven by noise. Under Assumption 2, a simulation, which satisfies this condition, can easily be performed as depicted in Alg. 1. First, random parameters $\boldsymbol{\omega}^{(n)}$ and $\boldsymbol{\psi}^{(n)}$ are drawn from the parameter probability distributions $\mathbb{P}_\omega$ and $\mathbb{P}_\psi$. Substituting these samples in our stochastic model, we obtain a deterministic system $\boldsymbol{F}(\cdot, \boldsymbol{\omega}^{(n)})$, $\boldsymbol{H}(\cdot, \boldsymbol{\psi}^{(n)})$. In order to initialize the simulation, we draw an initial state $\boldsymbol{x}_0$ from the corresponding distribution $\mathbb{P}_{\boldsymbol{x}_0}$. By iteratively drawing perturbations $\boldsymbol{\epsilon}_k^{(n)}$, $\boldsymbol{\eta}_k^{(n)}$ from $\mathbb{P}_\epsilon$, $\mathbb{P}_\eta$ and computing a single prediction step, sample trajectories $\boldsymbol{X}_{[0:K]}^{(n)}$, $\boldsymbol{Y}_{[0:K]}^{(n)}$ of the random field model $\{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})\}_{\boldsymbol{x} \in \mathbb{X}}$, $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x} \in \mathbb{X}}$ are computed. Finally, this procedure is repeated $N \in \mathbb{N}$ times in order to increase the statistical significance of the simulations.

B. Trajectory Sampling from Conditional Distributions

If we cannot draw samples from the parameter distributions $\mathbb{P}_\omega$, $\mathbb{P}_\psi$, e.g., in the case of Gaussian process models with infinitely many parameters, we have to sequentially draw the samples of the random variables $\boldsymbol{F}_{\boldsymbol{x}_k} = \boldsymbol{F}(\boldsymbol{x}_k, \boldsymbol{\omega})$, $\boldsymbol{H}_{\boldsymbol{x}_k} = \boldsymbol{H}(\boldsymbol{x}_k, \boldsymbol{\psi})$ in each prediction step k. Since this problem is identical for the state transition model $\{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})\}_{\boldsymbol{x} \in \mathbb{X}}$ and the observation model $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x} \in \mathbb{X}}$, we focus on the state transition model in the following.

Algorithm 2: Sampling from conditional distributions

```
Function sample_C($N$, $\mathbb{P}_\epsilon$, $\mathbb{P}_\eta$, $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_0},\dots,\boldsymbol{F}_{\boldsymbol{x}_{K-1}}}$,
  $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_0},\dots,\boldsymbol{H}_{\boldsymbol{x}_K}}$, $\mathbb{P}_{\boldsymbol{x}_0}$):
    for $n = 1, \dots, N$ do
        Sample $\boldsymbol{x}_0^{(n)}$ from $\mathbb{P}_{\boldsymbol{x}_0}$
        for $k = 0, \dots, K-1$ do
            Sample $\boldsymbol{F}_{\boldsymbol{x}_k}^{(n)}$ from $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}}^{(n)},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}^{(n)}}$
            Sample $\boldsymbol{H}_{\boldsymbol{x}_k}^{(n)}$ from $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_k}|\boldsymbol{H}_{\boldsymbol{x}_{k-1}}^{(n)},\dots,\boldsymbol{H}_{\boldsymbol{x}_0}^{(n)}}$
            Sample $\boldsymbol{\epsilon}_k^{(n)}$ and $\boldsymbol{\eta}_k^{(n)}$ from $\mathbb{P}_\epsilon$ and $\mathbb{P}_\eta$
            $\boldsymbol{Y}_{[k]}^{(n)} \leftarrow \boldsymbol{H}_{\boldsymbol{x}_k}^{(n)} + \boldsymbol{\eta}_k^{(n)}$
            $\boldsymbol{x}_{k+1}^{(n)} \leftarrow \boldsymbol{F}_{\boldsymbol{x}_k}^{(n)} + \boldsymbol{\epsilon}_k^{(n)}$
            Update $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}}^{(n)},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}^{(n)}}$, $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_k}|\boldsymbol{H}_{\boldsymbol{x}_{k-1}}^{(n)},\dots,\boldsymbol{H}_{\boldsymbol{x}_0}^{(n)}}$
        Sample $\boldsymbol{H}_{\boldsymbol{x}_K}^{(n)}$ from $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_K}|\boldsymbol{H}_{\boldsymbol{x}_{K-1}}^{(n)},\dots,\boldsymbol{H}_{\boldsymbol{x}_0}^{(n)}}$
        Sample $\boldsymbol{\eta}_K^{(n)}$ from $\mathbb{P}_\eta$
        $\boldsymbol{Y}_{[K]}^{(n)} \leftarrow \boldsymbol{H}_{\boldsymbol{x}_K}^{(n)} + \boldsymbol{\eta}_K^{(n)}$
```

The state transitions $\boldsymbol{F}_{\boldsymbol{x}_k}$ cannot be drawn independently in general, since they follow the joint distribution

$$\boldsymbol{F}(\boldsymbol{x}_0, \boldsymbol{\omega}), \dots, \boldsymbol{F}(\boldsymbol{x}_{K-1}, \boldsymbol{\omega}) \sim \mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_0},\dots,\boldsymbol{F}_{\boldsymbol{x}_{K-1}}} \tag{7}$$

with

$$\begin{aligned}&\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_0},\dots,\boldsymbol{F}_{\boldsymbol{x}_{K-1}}}(\mathbb{S}_1, \dots, \mathbb{S}_K) := \\ &\mathbb{P}_\omega(\boldsymbol{\omega} \in \Omega \colon \boldsymbol{F}(\boldsymbol{x}_0, \boldsymbol{\omega}) \in \mathbb{S}_1 \wedge \dots \wedge \boldsymbol{F}(\boldsymbol{x}_{K-1}, \boldsymbol{\omega}) \in \mathbb{S}_K)\end{aligned} \tag{8}$$

for any sets $\mathbb{S}_k \subset \mathbb{X}$, $k = 1, \dots, K$. Hence, independent sampling would lead to an inconsistent behavior with respect to Assumption 1, as the computed sequence of states would not be generated by a system of the form (1). However, this joint probability distribution can be expressed based on the Bayesian updates $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}}$ through

$$\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_0},\dots,\boldsymbol{F}_{\boldsymbol{x}_{K-1}}} = \prod_{k=0}^{K-1} \mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}}. \tag{9}$$

Therefore, we can iteratively sample $\boldsymbol{F}_{\boldsymbol{x}_k}$ from the conditional probabilities $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}}$, such that the samples satisfy (7). Furthermore, we can pursue the same procedure to obtain samples for the observation model $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x} \in \mathbb{X}}$ due to the equivalence of both sampling problems.

Under Assumption 3, we can exploit this iterative sampling in Alg. 2 to sample N random trajectories $\boldsymbol{Y}_{[0:K]}^{(n)}$. In contrast to Alg. 1, Alg. 2 takes as input the joint probability distributions $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_0},\dots,\boldsymbol{F}_{\boldsymbol{x}_{K-1}}}$, $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_0},\dots,\boldsymbol{H}_{\boldsymbol{x}_K}}$. Based on these densities, it iteratively samples $\boldsymbol{F}_{\boldsymbol{x}_k}^{(n)}$, $\boldsymbol{H}_{\boldsymbol{x}_k}^{(n)}$ from the conditional distributions $\mathbb{P}_{\boldsymbol{F}_{\boldsymbol{x}_k}|\boldsymbol{F}_{\boldsymbol{x}_{k-1}}^{(n)},\dots,\boldsymbol{F}_{\boldsymbol{x}_0}^{(n)}}$, $\mathbb{P}_{\boldsymbol{H}_{\boldsymbol{x}_k}|\boldsymbol{H}_{\boldsymbol{x}_{k-1}}^{(n)},\dots,\boldsymbol{H}_{\boldsymbol{x}_0}^{(n)}}$ at every prediction step k. By adding random noise to these values, the next state $\boldsymbol{x}_{k+1}$ and output $\boldsymbol{y}_k$ are obtained. Finally, it computes the Bayesian updates of the conditional distributions. Thereby, Alg. 2 iteratively constructs trajectories $\boldsymbol{X}_{[0:K]}^{(n)}$, $\boldsymbol{Y}_{[0:K]}^{(n)}$, which are consistent with Assumption 1.

C. Probabilistic Error Bounds for Trajectory Simulation

We want to exploit the sample trajectories introduced in the previous sections to derive confidence levels of sets $\mathbb{S}_k$ and $\mathbb{T}_k$, $k = 0, \ldots, K$, for trajectories of the unknown system (1). We interpret each sample trajectory as a single sample and approximate the probability (2) empirically using (4). Since this estimate depends on random trajectories, it is a random variable itself, and we need to bound the deviation of the estimate from the true probability. However, we first define the set analogue of trajectories in order to have an intuitive notation.

Definition 3: A trajectory of sets $\mathbb{S}_{[0:K]} \subset \mathbb{X} \times \ldots \times \mathbb{X}$ is defined such that $\mathbb{S}_{[k]} = \mathbb{S}_k$ for all $k = 0, \ldots, K$ and a sequence of confidence regions $\mathbb{S}_k$. Furthermore, we extend the element notation "$\in$" such that

$$\boldsymbol{X}_{[0:K]} \in \mathbb{S}_{[0:K]} \quad \Leftrightarrow \quad \boldsymbol{X}_{[k]} \in \mathbb{S}_{[k]} \; \forall k = 0, \ldots, K. \tag{10}$$

Although not stating it here explicitly, we can equivalently define set trajectories for output confidence sets $\mathbb{T}_k$. Moreover, the extension of the element notation directly extends the indicator function defined in (6), such that we can express (2) via trajectories. We exploit this in the following lemma.

Lemma 1: Consider a dynamical system (1) and a stochastic model $\{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})\}_{\boldsymbol{x} \in \mathbb{X}}$, $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x} \in \mathbb{X}}$ satisfying Assumption 1. If the model is used to sample N trajectories $\boldsymbol{X}^{(n)}_{[0:K]}$, $\boldsymbol{Y}^{(n)}_{[0:K]}$, then, for all trajectories of confidence regions $\mathbb{S}_{[0:K]}$, $\mathbb{T}_{[0:K]}$ with $\mathbb{S}_{[k]} \subseteq \mathbb{X}$, $\mathbb{T}_{[k]} \subseteq \mathbb{Y}$, $k = 0, \ldots, K$, and parameters $t \in [0, 1]$, an unknown system trajectory $\boldsymbol{X}_{[0:K]}$, $\boldsymbol{Y}_{[0:K]}$ satisfies

$$\mathbb{P}\left(\boldsymbol{X}_{[0:K]} \in \mathbb{S}_{[0:K]}, \boldsymbol{Y}_{[0:K]} \in \mathbb{T}_{[0:K]}\right) \geq 1 - \delta \tag{11}$$

with confidence level

$$\delta = 1 - \left(\hat{P} - t\right)\left(1 - \mathrm{e}^{-2Nt^2}\right) \tag{12}$$

$$\hat{P} = \frac{1}{N} \sum_{n=1}^{N} I_{\mathbb{S}_{[0:K]}, \mathbb{T}_{[0:K]}}\left(\boldsymbol{X}^{(n)}_{[0:K]}, \boldsymbol{Y}^{(n)}_{[0:K]}\right). \tag{13}$$

Proof: For notational convenience, we denote the true probability of confidence sets $\mathbb{S}_{[0:K]}$ and $\mathbb{T}_{[0:K]}$ as

$$P := \mathbb{P}\left(\boldsymbol{X}_{[0:K]} \in \mathbb{S}_{[0:K]}, \boldsymbol{Y}_{[0:K]} \in \mathbb{T}_{[0:K]}\right). \tag{14}$$

The approximate probability $\hat{P}$, which is defined analogously to (4), is a random variable, as it depends on the randomly sampled trajectories $\boldsymbol{X}^{(n)}_{[0:K]}$, $\boldsymbol{Y}^{(n)}_{[0:K]}$. However, it is well known that its expectation equals the true probability P, i.e.,

$$\mathbb{E}\left[\hat{P}\right] = P.$$

Therefore, we can apply Hoeffding's inequality to obtain

$$\mathbb{P}\left(P \leq \hat{P} - t\right) \leq \mathrm{e}^{-2Nt^2} \tag{15}$$

for all $t \in [0, 1]$. Furthermore, we can express the true probability P through conditional probabilities as

$$\begin{aligned} P = {} & \mathbb{P}\left(\boldsymbol{X}_{[0:K]} \in \mathbb{S}_{[0:K]}, \boldsymbol{Y}_{[0:K]} \in \mathbb{T}_{[0:K]} \middle| P > \hat{P} - t\right) \mathbb{P}\left(P > \hat{P} - t\right) \\ & + \mathbb{P}\left(\boldsymbol{X}_{[0:K]} \in \mathbb{S}_{[0:K]}, \boldsymbol{Y}_{[0:K]} \in \mathbb{T}_{[0:K]} \middle| P \leq \hat{P} - t\right) \mathbb{P}\left(P \leq \hat{P} - t\right). \end{aligned}$$

If we employ the trivial bounds

$$\begin{aligned} 0 &\leq \mathbb{P}\left(P \leq \hat{P} - t\right) \\ \hat{P} - t &\leq \mathbb{P}\left(\boldsymbol{X}_{[0:K]} \in \mathbb{S}_{[0:K]}, \boldsymbol{Y}_{[0:K]} \in \mathbb{T}_{[0:K]} \middle| P > \hat{P} - t\right) \end{aligned}$$

and express $\mathbb{P}(P > \hat{P} - t)$ through (15) using the complimentary event, then we obtain

$$P \geq (\hat{P} - t)\left(1 - \mathrm{e}^{-2Nt^2}\right).$$

■

Lemma 1 allows to calculate the probability of all confidence set trajectories $\mathbb{S}_{[0:K]}$, $\mathbb{T}_{[0:K]}$ based on N random trajectories. However, it contains a parameter t, which has to be tuned for maximum probability. Furthermore, it gives no information about the number N of sample trajectories required to achieve a desired probability δ. This is addressed in the following theorem.

Theorem 1: Consider a dynamical system (1) and a stochastic model $\{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})\}_{\boldsymbol{x} \in \mathbb{X}}$, $\{\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{\psi})\}_{\boldsymbol{x} \in \mathbb{X}}$ satisfying Assumption 1, as well as a desired confidence level $\bar{\delta} \in (0, 0.19]$. If we sample more than

$$N \geq \left(\frac{11}{10\sqrt{2}\bar{\delta}}\right)^3 \tag{16}$$

trajectories $\boldsymbol{X}^{(n)}_{[0:K]}$,$\boldsymbol{Y}^{(n)}_{[0:K]}$ and if the chosen trajectories of confidence regions $\mathbb{S}_{[0:K]}$, $\mathbb{T}_{[0:K]}$ with $\mathbb{S}_{[k]} \subseteq \mathbb{X}$, $\mathbb{T}_{[k]} \subseteq \mathbb{Y}$, $k = 0, \ldots, K$, satisfy

$$\frac{1}{N} \sum_{n=1}^{N} I_{\mathbb{S}_{[0:K]}, \mathbb{T}_{[0:K]}}\left(\boldsymbol{X}^{(n)}_{[0:K]}, \boldsymbol{Y}^{(n)}_{[0:K]}\right) = 1, \tag{17}$$

then we can guarantee that (11) holds with $\delta \leq \bar{\delta}$.

Proof: Define

$$t(N) = \frac{1}{\sqrt{2}} N^{-\frac{1}{3}}.$$

Then, we have

$$\begin{aligned} \delta &= \frac{1}{\sqrt{2}} N^{-\frac{1}{3}} + \mathrm{e}^{-N^{\frac{1}{3}}} - \frac{1}{\sqrt{2}} N^{-\frac{1}{3}} \mathrm{e}^{-N^{\frac{1}{3}}} \\ &\leq \frac{1}{\sqrt{2}} N^{-\frac{1}{3}} + \mathrm{e}^{-N^{\frac{1}{3}}}, \end{aligned} \tag{18}$$

where the second line follows from the nonnegativity of all summands. If we ensure $N \geq \bar{N}$ with $\bar{N}$ implicitly defined by

$$100\mathrm{e}^{-\bar{N}^{\frac{1}{3}}} \leq 5\sqrt{2}\bar{N}^{-\frac{1}{3}}, \tag{19}$$

then (18) simplifies to $100\delta \leq 55\sqrt{2}N^{-1/3}$ for all $N \geq \bar{N}$. Therefore,

$$N \geq \left(\frac{55\sqrt{2}}{100\bar{\delta}}\right)^3$$

implies $\delta \leq \bar{\delta}$, if $N \geq \bar{N}$. It remains to explicitly determine $\bar{N}$ and transfer it to a condition for the allowed range of $\bar{\delta}$. For the derivation of $\bar{N}$ we rewrite (19) as

$$-N^{\frac{1}{3}} \mathrm{e}^{-N^{\frac{1}{3}}} \geq -\frac{5\sqrt{2}}{100}.$$

The left-hand side corresponds to the definition of the Lambert W function, such that we obtain

$$\bar{N} \geq \left(-W_{-1}\left(-\frac{5\sqrt{2}}{100}\right)\right).$$

Finally, we substitute this value into (18), which results in

$$\bar{\delta} \leq 0.19 \leq \frac{55\sqrt{2}}{100}\bar{N}^{-\frac{1}{3}}.$$

■

We fixed the estimated probability $\hat{P}$ in Theorem 1 in order to simplify the presentation. It is also possible to calculate the required number $\bar{N}$ of sampled trajectories for $\hat{P} < 1$. However, the interval for the worst case confidence level $\bar{\delta}$ must be adapted accordingly. Furthermore, the parameter t is not chosen optimally in the proof of Theorem 1, but such that it allows the computation of a sufficient lower bound for the required number of samples $\bar{N}$. Therefore, we can expect that there exists a better value of t for each N than the one used in the proof. Although there is no simple analytic expression for the optimal t, it is possible to obtain it through convex optimization as shown in the following theorem.

Theorem 2: Consider a fixed $N \in \mathbb{N}$. Then, the global minimum of (12) with respect to t can be obtained through convex optimization constrained to the interval $[\underline{t}, 1]$ with

$$\underline{t} = \frac{((1-\sqrt{-3})(-4N\hat{P}^2-9))}{12\sqrt[3]{\theta}} - \frac{(1+\sqrt{-3})\sqrt[3]{\theta}}{12N} + \frac{\hat{P}}{3} \tag{20}$$

$$\theta = 8N^3\hat{P}^3 + \sqrt{27\left(-16N^5\hat{P}^4 - 36N^4\hat{P}^2 - 27N^3\right)}. \tag{21}$$

Proof: It is trivial to derive that the second derivative of (12) with respect to t equals zero only if

$$\hat{P} - 3t - 4\hat{P}Nt^2 + 4Nt^3 = 0.$$

Applying Cardano's method to this problem, we find that the only zero $\underline{t}$ in the interval $[0, 1]$ is given by (20). Since the second derivative of (12) is positive at $t = 1$, it follows that it is nonnegative on the whole interval $[\underline{t}, 1]$. Hence, (12) is convex with respect to t on the interval $[\underline{t}, 1]$. It remains to show that the global optimum indeed lies in this interval. This holds, because the first derivative of (12) with respect to t equals 0 at $t = 0$. Due to concavity in the interval $[0, \underline{t})$, the derivative cannot become 0 at any other point in this interval. Since $t = 0$ leads to the local maximum $\delta = 1$, the global minimum cannot be in the interval $[0, \underline{t})$ concluding the proof. ■

V. Numerical Evaluation

A. Bayesian Hidden Markov Models

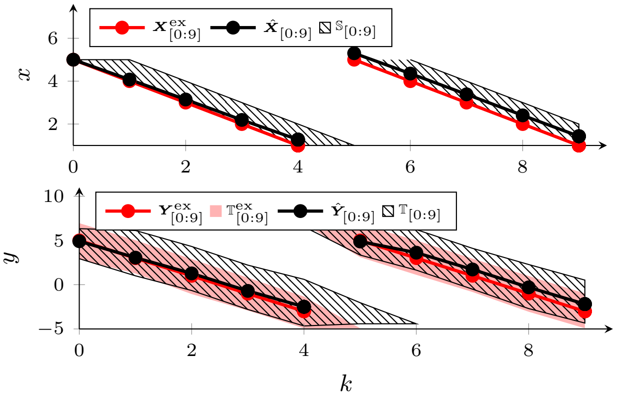

Fig. 1. Trajectories of confidence regions for the learned HMM (hatched) are only slightly larger than those for the exact system (red).

We demonstrate the flexibility of our approach by applying it to hidden Markov models.[2] For this reason, we decompose a discrete state Markov system into the structure of (1), learn a hidden Markov model with Bayesian methods following [17], and construct confidence intervals for this model. We consider an unknown hidden Markov system with state transition matrix

$$\boldsymbol{A} = \begin{bmatrix} 0.05 & 0 & 0 & 0 & 0.95 \\ 0.95 & 0.05 & 0 & 0 & 0 \\ 0 & 0.95 & 0.05 & 0 & 0 \\ 0 & 0 & 0.95 & 0.05 & 0 \\ 0 & 0 & 0 & 0.95 & 0.05 \end{bmatrix} \tag{22}$$

and conditional normal output data $y|x \sim \mathcal{N}(2x-5, \sigma)$ with $\sigma = 0.5$. We can equivalently express this system as

$$f(x) = \operatorname{mod}(x-2, 5) + 1 \tag{23}$$

$$h(x) = 2x - 5 \tag{24}$$

with process noise

$$\mathbb{P}_\epsilon(\epsilon) = \begin{cases} 0.95 & \text{if } \epsilon = 0 \\ 0.05 & \text{if } \epsilon = 1 \end{cases} \tag{25}$$

and observation noise $\eta \sim \mathcal{N}(0, \sigma)$. We follow the approach introduced in [17] to define the prior distribution and to train the Bayesian hidden Markov model with 1000 training samples. We directly learn $\boldsymbol{A}$, i.e., a joint model for $f(\cdot)$ and $\mathbb{P}_\epsilon$. The calculation of posterior distributions is performed numerically using Markov chain Monte Carlo, where the first 1000 samples are discarded as burn-in phase [17].

We compare trajectories of confidence regions for the exact system and the learned hidden Markov model over a horizon of $K = 9$ steps, starting at a fixed initial state $x_0 = 5$. In order to determine the confidence region, we sample 1000 trajectories with Alg. 1 and define $\mathbb{S}_{[0:9]}$ and $\mathbb{T}_{[0:9]}$, such that they contain approximately 60% of the trajectories. Based on Lemma 1 and Theorem 2, we compute the confidence levels for the exact system and the model as $\delta_{\text{exact}} = 0.39$ and $\delta = 0.43$ using $N = 10000$ sample trajectories. The confidence sets $\mathbb{T}_{[0:9]}$, $\mathbb{S}_{[0:9]}$, $\mathbb{S}^{\text{ex}}_{[0:9]}$ and mean trajectories $\hat{\boldsymbol{X}}_{[0:9]}$, $\hat{\boldsymbol{Y}}_{[0:9]}$, $\boldsymbol{X}^{\text{ex}}_{[0:9]}$, $\boldsymbol{Y}^{\text{ex}}_{[0:9]}$ of the Bayesian hidden Markov model and the exact system are depicted in Fig. 1. Note that the trajectory of confidence regions of the exact system is only slightly tighter than for the hidden Markov model. However, in the case of the state trajectory, the confidence regions of the exact system correspond to the mean trajectory $\boldsymbol{X}^{\text{ex}}_{[0:9]}$, such that we refrain from plotting the trajectory of confidence regions.

[2] The code is available at https://gitlab.lrz.de/alederer/MC4LPM

B. Gaussian Process Model of the Inverted Pendulum

In our second example we consider the inverted pendulum

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} x_2 \\ \frac{g}{l}\sin(x_1) + \frac{c}{ml^2}x_2 \end{bmatrix} \tag{26}$$

$$\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{x} \tag{27}$$

with parameters $g = 9.81$, $l = 0.5$, $m = 0.15$ and $c = 0.1$ as described in [8]. We consider a normally distributed observation noise $\eta \sim \mathcal{N}(0, \sigma)$ with standard deviation $\sigma = 0.05$ and no process noise $\epsilon = 0$. We train two Gaussian processes with squared exponential kernels independently for each dimension based on 5 trajectories with 100 training pairs $(\boldsymbol{x}, y_i)$, $i = 1, 2$. The Gaussian process hyperparameters are obtained through log-likelihood maximization. Our goal is the computation of confidence regions for state trajectories $\boldsymbol{X}_{[0:100]}$ generated by the deterministic process $\boldsymbol{f}(\cdot)$ over a prediction horizon $K = 100$. Therefore, we assume $\mathbb{T}_{[k]} = \mathbb{Y}$, $\forall k = 0, \ldots, 100$ in the following.

In order to determine a confidence region, we sample 100 trajectories with Alg. 2. Each set $\mathbb{S}_{[k]}$ is defined such that it robustly contains all states $\boldsymbol{X}_{[k]}^{(n)}$, $n = 0, \ldots, 100$. Assuming $\hat{P} = 1$ and setting $\bar{\delta} = 0.05$, it is necessary to sample $N = 3765$ trajectories due to Theorem 1. The sample trajectories $\boldsymbol{X}_{[0:100]}^{(n)}$ led to an empirical probability of $\hat{P} = 1$ in our simulations, such that we obtain a confidence level of $\delta = 0.029$ based on Lemma 1 and Theorem 2. The resulting trajectory of confidence regions $\mathbb{S}_{[0:100]}$ and mean trajectory $\hat{\boldsymbol{X}}_{[0:100]}$ are illustrated for both states in Fig. 2.

We compare our sampling-based approach to the robust model approach based on high probability uniform error bounds in [8]. We set $\beta^2 = 2$ because the $\pm 2\sigma(\boldsymbol{x})$ interval around the Gaussian process mean has probability of 0.95 for each fixed state $\boldsymbol{x}$. While this choice of β is below the theoretically required value to obtain confidence regions with probability of at least 0.95, it is a less conservative choice. Furthermore, we determine a Lipschitz constant $L_f = 1.7$ through numerical optimization, which also does not correspond to a conservative choice. However, as clearly shown in Fig. 2, the resulting confidence regions $\mathbb{E}_{[0:100]}$ grow quickly and are larger than our confidence sets $\mathbb{S}_{[0:100]}$, although the mean trajectory $\boldsymbol{X}_{[0:100]}^{\mathrm{rob}}$ is almost equal to our prediction $\hat{\boldsymbol{X}}_{[0:100]}$.

VI. Conclusion

In this paper, we present a novel method to determine confidence sets and levels for trajectories predicted with learned probabilistic models. Based on a Monte Carlo simulation of the system, we calculate confidence levels of sets containing the predicted trajectory. Due to the highly nonlinear dependence on the number of random samples, we derive a sufficient number of samples to achieve a desired confidence level. The flexibility and superior performance compared to existing approaches is demonstrated in simulations of a Bayesian hidden Markov model and a Gaussian process model.

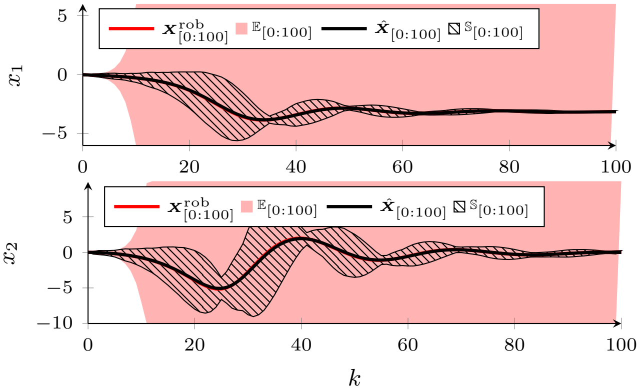

Fig. 2. Confidence regions obtained by our sampling method (hatched) are significantly tighter than confidence regions based on the robust model approach (red) introduced in [8].

References

[1] Z. Ghahramani, "Probabilistic Machine Learning and Artificial Intelligence," *Nature*, vol. 27, pp. 452–459, 2015.

[2] A. Lederer, J. Umlauft, and S. Hirche, "Uniform Error Bounds for Gaussian Process Regression with Application to Safe Control," in *Advances in Neural Information Processing Systems*, 2019.

[3] K. Chua, R. Calandra, R. McAllister, and S. Levine, "Deep Reinforcement Learning in a Handful of Trials using Probabilistic Dynamics Models," in *Advances in Neural Information Processing Systems*, 2018, pp. 4754–4765.

[4] T. Qu, S. Yu, Z. Shi, and H. Chen, "Modeling Driver's Car-following Behavior based on Hidden Markov Model and Model Predictive Control: A Cyber-physical System Approach," in *Proceedings of the Asian Control Conference*, 2018, pp. 114–119.

[5] J. Umlauft, T. Beckers, and S. Hirche, "Scenario-based Optimal Control for Gaussian Process State Space Models," in *Proceedings of the European Control Conference*, 2018.

[6] S. Depeweg, J. M. Hernández-Lobato, F. Doshi-Velez, and S. Udluft, "Learning and Policy Search in Stochastic Dynamical Systems with Bayesian Neural Networks," in *International Conference on Learning Representations*, 2017, pp. 1–14.

[7] A. Capone and S. Hirche, "Anticipating the Long-Term Effect of Online Learning in Control," in *Proceedings of the American Control Conference*, 2020.

[8] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based Model Predictive Control for Safe Exploration," in *Proceedings of the IEEE Conference on Decision and Control*, 2018, pp. 6059–6066.

[9] T. Beckers, J. Umlauft, and S. Hirche, "Mean Square Prediction Error of Misspecified Gaussian Process Models," in *Proceedings of the IEEE Conference on Decision and Control*, 2018, pp. 1162–1167.

[10] A. Girard, C. E. Rasmussen, J. Q. Candela, and R. Murray-Smith, "Gaussian Process Priors with Uncertain Inputs-Application to Multiple-Step Ahead Time Series Forecasting," *Advances in Neural Information Processing Systems*, pp. 545–552, 2003.

[11] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight Uncertainty in Neural Networks," in *Proceedings of the International Conference on Machine Learning*, 2015, pp. 1613–1622.

[12] S. L. Scott, "Bayesian Methods for Hidden Markov Models: Recursive Computing in the 21st Century," *Journal of the American Statistical Association*, vol. 97, no. 457, pp. 337–351, 2002.

[13] W. Gilks, S. Richardson, and D. Spiegelhalter, Eds., *Markov Chain Monte Carlo in Practice*. New York: Chapman & Hall, 1995.

[14] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: The MIT Press, 2006.

[15] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, 2nd ed. New York, NY: Springer Science+Business Media, 2004.

[16] R. Bakx, R. H. P. Kleiss, and F. Versteegen, "First- and Second-order Error Estimates in Monte Carlo Integration," *Computer Physics Communications*, vol. 208, pp. 29–34, 2016.

[17] T. Rydén, "EM versus Markov Chain Monte Carlo for Estimation of Hidden Markov Models: A Computational Perspective," *Bayesian Analysis*, vol. 3, no. 4, pp. 659–688, 2008.