Proceedings of Machine Learning Research vol xxx:1–15, 2021

The Value of Data in Learning-Based Control for Training Subset Selection

Armin Lederer ARMIN.LEDERER@TUM.DE
Alexandre Capone ALEXANDRE.CAPONE@TUM.DE
Thomas Beckers T.BECKERS@TUM.DE
Jonas Umlauft JONAS.UMLAUFT@TUM.DE
Sandra Hirche HIRCHE@TUM.DE
Chair of Information-oriented Control
Department of Electrical and Computer Engineering
Technical University of Munich
D-80333 Munich, Germany

Abstract

Despite the existence of formal guarantees for learning-based control approaches, the relationship between data and control performance is still poorly understood. In this paper, we present a measure to quantify the value of data within the context of a predefined control task. Our approach is applicable to a wide variety of unknown nonlinear systems that are to be controlled by a generic learning-based control law. We model the unknown component of the system using Gaussian processes, which in turn allows us to directly assess the impact of model uncertainty on control. Results obtained in numerical simulations indicate the efficacy of the proposed measure.

Keywords: data-driven control, Gaussian processes, data-efficient learning, safe learning-based control

1. Introduction

Learning-based control is rapidly becoming an attractive alternative to traditional control approaches, particularly in settings with limited knowledge or prohibitive system complexity (Deisenroth et al., 2015; Chua et al., 2018). This has prompted a vast amount of research into the theoretical properties of control approaches based on nonparametric and probabilistic models obtained from supervised machine learning, yielding techniques that guarantee either safety or performance requirements (Aswani et al., 2013; Beckers et al., 2019; Fisac et al., 2019; Capone and Hirche, 2019; Gahlawat et al., 2020; Lederer et al., 2020).

While theoretical guarantees for control performance can be obtained in various settings, the direct relationship between collected data and control performance in learning-based control with nonparametric, probabilistic models is still poorly understood. In experimental design, the value of data is often quantified using information theoretical quantities, such as mutual information or entropy (Pukelsheim, 2006). Although these quantities have been used extensively to guide exploration and control strategies (Alpcan and Shames, 2015; Koller et al., 2018; Capone et al., 2020; Umlauft et al., 2020), they do not provide direct insight into the impact of data on control performance.

In recent years, a handful of efforts has been carried out towards understanding the influence of collected data on control performance. Lederer et al. (2021) have proposed a measure for quantifying the value of data points with respect to a specific control task based on Gaussian process priors.

In a similar vein, Capone et al. (2021) have developed an algorithm to identify the most useful data points for successfully performing multiple control tasks. However, the technique presented in Lederer et al. (2021) is only applicable to a restricted class of systems, and the method of Capone et al. (2021) exhibits poor scalability.

In this work, we present a measure for quantifying the value that training data has on the performance of learning-based control of nonlinear systems. The proposed technique employs Gaussian process regression to quantify model uncertainty, and subsequently derive the proposed information measure. Our work generalizes the results from Lederer et al. (2021) by considering a markedly richer class of systems.

The remainder of this paper is structured as follows. In Section 2, we formally state the considered problem. In Section 3, we briefly discuss how Gaussian processes are employed to model the unknown system, and present some preliminary results. Afterwards, in Section 4, we derive the proposed information measure. We then discuss strategies for data selection, in Section 5, after which we present some experimental results, in Section 6. Finally, some concluding remarks are provided in Section 7.

2. Problem Statement

We consider a dynamical system

$$\dot{\boldsymbol{x}} = \boldsymbol{g}(\boldsymbol{z}) = \boldsymbol{A}\boldsymbol{f}(\boldsymbol{z}), \tag{1}$$

where $\boldsymbol{z} = \begin{bmatrix}\boldsymbol{x}^T & \boldsymbol{u}^T\end{bmatrix}^T \in \mathbb{R}^{d_z}$, $d_z = d_x + d_u$, is the concatenation of the state $\boldsymbol{x} \in \mathbb{X}$ and the input $\boldsymbol{u} \in \mathbb{U}$ for compact sets $\mathbb{X} \subset \mathbb{R}^{d_x}$ and $\mathbb{U} \subset \mathbb{R}^{d_u}$. We assume to know the matrix $\boldsymbol{A} \in \mathbb{R}^{d_x \times d_f}$, whereas the function $\boldsymbol{f} : \mathbb{R}^{d_x+d_u} \to \mathbb{R}^{d_f}$ is unknown. This system structure is very flexible and allows general multi-dimensional nonlinear systems ($\boldsymbol{A} = \boldsymbol{I}_{d_x}$, $d_f = d_x$), as well as correlation in the outputs, which can be found, e.g., in Euler-Lagrange systems due to the symmetry of the mass matrix. We assume to have an approximate model $\hat{\boldsymbol{f}} : \mathbb{R}^{d_x+d_u} \to \mathbb{R}^{d_f}$ of the unknown function $\boldsymbol{f}(\cdot, \cdot)$, which is often available in practice, as well as measurement data. The assumptions on the measurement data are formalized as follows.

Assumption 1 *A data set containing N measurement pairs*

$$\mathbb{D}_N = \left\{ \boldsymbol{z}^{(n)} := \begin{bmatrix}\boldsymbol{x}^{(n)} \\ \boldsymbol{u}^{(n)}\end{bmatrix}, \boldsymbol{y}^{(n)} = \boldsymbol{g}\left(\boldsymbol{z}^{(n)}\right) + \boldsymbol{\epsilon}^{(n)} \right\}_{n=1}^{N} \tag{2}$$

is available, where $\boldsymbol{\epsilon}^{(n)} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\mathrm{on}})$ is i.i.d. Gaussian noise with covariance matrix $\boldsymbol{\Sigma}_{\mathrm{on}}$.

Additionally, we assume that the unknown functions $f_i(\cdot)$ are well behaved, as expressed in the following.

Assumption 2 *The unknown functions $f_i(\cdot)$ are Lipschitz continuous with Lipschitz constants L_{f_i}, $i = 1, \ldots, d_f$.*

We assume to have a learning-based control law $\boldsymbol{\pi} : \mathbb{X} \times \left(\mathbb{X} \times \mathbb{U} \times \mathbb{R}^{d_x}\right)^N \times \mathbb{R}_{0,+} \to \mathbb{U}$ that causally maps a state $\boldsymbol{x}$ to a control input $\boldsymbol{u}$ depending on the previously observed training data $\mathbb{D}_N$ and the current time $t \in \mathbb{R}_{0,+}$. The control law is designed to achieve a control task, e.g., stabilization with respect to a reference point, or tracking of a reference trajectory. The effectiveness of the control

law with respect to this task is measured via a Lyapunov function[1] $V : \mathbb{X} \times \mathbb{R}_{0,+} \to \mathbb{R}_{0,+}$ and its temporal derivative $\dot{V}(\cdot,\cdot)$ along trajectories of the closed loop system

$$\dot{\boldsymbol{x}} = \tilde{\boldsymbol{g}}(\boldsymbol{x}) = \boldsymbol{g}\left(\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\pi}(\boldsymbol{x}, \boldsymbol{x}^{(1)}, \boldsymbol{u}^{(1)}, \ldots, t) \end{bmatrix}\right). \tag{3}$$

Since we do not know the function $\boldsymbol{f}(\cdot)$ but only have a training data set $\mathbb{D}_N$ and an approximate model $\hat{\boldsymbol{f}}(\cdot)$, the warranted control performance, measured through the size of the region in which $\dot{V}(\boldsymbol{x}) \leq 0$ is not guaranteed, strongly depends on the training data. In such a setting, it is crucial to understand the interrelation between training data and control performance, e.g., for determining where additional training data should be acquired to increase control performance, or when a subset of the training data must be selected to reduce the computational complexity of learning. Therefore, we consider the problem of determining the value of data in learning-based control.

3. Gaussian Process Regression

For determining the value of data in control, we employ a Gaussian process (GP) formulation, such that we can relate the control performance to the model uncertainty. We first introduce the foundations of Gaussian process regression in Section 3.1, before we explain the extension to multiple outputs, in Section 3.2. Finally, we propose an output-decoupling formulation using linear models of coregionalization and propose a novel uniform error bound, in Section 3.3.

3.1. Single Output Gaussian Processes

A Gaussian process $\mathcal{GP}(\hat{f}(\cdot,), k(\cdot,\cdot))$, uniquely defined through a prior mean function $\hat{f} : \mathbb{R}^{d_z} \to \mathbb{R}$ and a covariance function $k : \mathbb{R}^{d_z} \times \mathbb{R}^{d_z} \to \mathbb{R}_{0,+}$, is a generalization of Gaussian distributions (Rasmussen and Williams, 2006). The prior mean function $\hat{f}(\cdot)$ is often used to include approximate models in the regression, whereas the covariance function describes prior assumptions on properties such as smoothness or periodicity. A commonly used covariance function, on which we also focus in the following analysis for clarity of exposition, is the squared exponential kernel

$$k(\boldsymbol{z}, \boldsymbol{z}') = s_f^2 \exp\left(-\sum_{i=1}^{d_z} \frac{(z_i - z_i')^2}{2l_i^2}\right), \tag{4}$$

where $s_f^2 \in \mathbb{R}_{0,+}$ and $l_i \in \mathbb{R}_+$ denote the signal variance and length scales, respectively.

In order to perform regression with the GP, we consider a scalar system (1) with $d_f = 1$. Then, the joint prior distribution of training targets $\boldsymbol{t} = \begin{bmatrix} y^{(1)} & \cdots & y^{(N)} \end{bmatrix}^T$ and the unknown function value $f(\boldsymbol{z})$ for input $\boldsymbol{z}$ is given by

$$\begin{bmatrix} \boldsymbol{t} \\ f(\boldsymbol{z}) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \hat{f}(\boldsymbol{Z}) \\ \hat{f}(\boldsymbol{z}) \end{bmatrix}, \begin{bmatrix} k(\boldsymbol{Z}, \boldsymbol{Z}) + \sigma_{\text{on}}^2 \boldsymbol{I}_N & k(\boldsymbol{Z}, \boldsymbol{z}) \\ k^T(\boldsymbol{Z}, \boldsymbol{z}) & k(\boldsymbol{z}, \boldsymbol{z}) \end{bmatrix}\right), \tag{5}$$

where we use the abbreviations $\hat{f}(\boldsymbol{Z}) \in \mathbb{R}^N$, $k(\boldsymbol{Z}, \boldsymbol{Z}) \in \mathbb{R}^{N \times N}$ and $k(\boldsymbol{Z}, \boldsymbol{Z}) \in \mathbb{R}^N$ with elements defined as $\hat{f}_n(\boldsymbol{Z}) = \hat{f}(\boldsymbol{z}^{(n)})$, $k_{n,n'}(\boldsymbol{Z}, \boldsymbol{Z}) = k(\boldsymbol{z}^{(n)}, \boldsymbol{z}^{(n')})$ and $k_n(\boldsymbol{Z}, \boldsymbol{z}) = k(\boldsymbol{z}^{(n)}, \boldsymbol{z})$, $n, n' = 1, \ldots, N$, respectively. By conditioning the GP on the training data, we obtain the posterior distribution

$$f(\boldsymbol{z}) | y^{(1)}, \ldots, y^{(N)}, \boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(N)}, \boldsymbol{z} \sim \mathcal{N}(\mu(\boldsymbol{z}), \sigma^2(\boldsymbol{z})) \tag{6}$$

1. A Lyapunov function $V : \mathbb{X} \times \mathbb{R}_{0,+} \to \mathbb{R}_{0,+}$ is positive definite, i.e., $V(\boldsymbol{x}, t) \geq 0$ with equality if and only if $\boldsymbol{x} = 0$.

with posterior mean and variance

$$\mu(\boldsymbol{z}) = \hat{f}(\boldsymbol{z}) + k^T(\boldsymbol{Z}, \boldsymbol{z}) \left(k(\boldsymbol{Z}, \boldsymbol{Z}) + \sigma_{\text{on}}^2 \boldsymbol{I}_N\right)^{-1} (\boldsymbol{t} - \hat{f}(\boldsymbol{Z})) \tag{7}$$

$$\sigma^2(\boldsymbol{z}) = k(\boldsymbol{z}, \boldsymbol{z}) - k^T(\boldsymbol{Z}, \boldsymbol{z}) \left(k(\boldsymbol{Z}, \boldsymbol{Z}) + \sigma_{\text{on}}^2 \boldsymbol{I}_N\right)^{-1} k(\boldsymbol{Z}, \boldsymbol{z}). \tag{8}$$

3.2. Multiple-Output Gaussian Process Regression

In order to apply Gaussian processes to multiple-output regression problems, we can proceed analogously to the single output case. For illustrative purposes, we assume for now that $\boldsymbol{A} = \boldsymbol{I}_{d_x}$ and $d_f = d_x$ in (1), such that we have noisy measurements of the functions $f_i(\cdot)$, $i = 1, \dots, d_f$ in the data set $\mathbb{D}_N$. We start again with the prior GP distribution

$$\boldsymbol{f}(\cdot) \sim \mathcal{GP}\left(\hat{\boldsymbol{f}}(\cdot), \boldsymbol{K}(\cdot, \cdot)\right), \tag{9}$$

where we have to consider a vector-valued prior mean function $\hat{\boldsymbol{f}} : \mathbb{R}^{d_z} \to \mathbb{R}^{d_f}$ and a matrix kernel function $\boldsymbol{K} : \mathbb{R}^{d_z} \times \mathbb{R}^{d_z} \to \mathbb{R}_{0,+}^{d_f \times d_f}$, in which each element $k_{m,m'} : \mathbb{R}^{d_z} \times \mathbb{R}^{d_z} \to \mathbb{R}_{0,+}$ is a kernel. By concatenating the training targets $\boldsymbol{y}_i^{(n)}$ in the vector $\boldsymbol{t}^T = [y_1^{(1)} \ \cdots \ y_1^{(N)} \ y_2^{(1)} \ \cdots \ y_{d_x}^{(N)}]$ and conditioning the joint distribution of $\boldsymbol{t}$ and $\boldsymbol{f}(\boldsymbol{z})$ on the training data $\mathbb{D}_N$, analogously to (6) we obtain a multivariate Gaussian distribution with mean and covariance matrix

$$\boldsymbol{\mu}(\boldsymbol{z}) = \hat{\boldsymbol{f}}(\boldsymbol{z}) + \boldsymbol{K}^T(\boldsymbol{Z}, \boldsymbol{z}) \left(\boldsymbol{K}(\boldsymbol{Z}, \boldsymbol{Z}) + \boldsymbol{\Sigma}_{\text{on}} \otimes \boldsymbol{I}_N\right)^{-1} \left(\boldsymbol{t} - \hat{\boldsymbol{f}}(\boldsymbol{Z})\right) \tag{10}$$

$$\boldsymbol{\Sigma}(\boldsymbol{z}) = \boldsymbol{K}(\boldsymbol{z}, \boldsymbol{z}) - \boldsymbol{K}^T(\boldsymbol{Z}, \boldsymbol{z}) \left(\boldsymbol{K}(\boldsymbol{Z}, \boldsymbol{Z}) + \boldsymbol{\Sigma}_{\text{on}} \otimes \boldsymbol{I}_N\right)^{-1} \boldsymbol{K}(\boldsymbol{Z}, \boldsymbol{z}), \tag{11}$$

where we extend the shorthand notation from Section 3.1 using

$$\boldsymbol{K}(\boldsymbol{Z}, \boldsymbol{Z}) = \begin{bmatrix} k_{1,1}(\boldsymbol{Z}, \boldsymbol{Z}) & \cdots & k_{1,d_f}(\boldsymbol{Z}, \boldsymbol{Z}) \\ \vdots & \ddots & \vdots \\ k_{d_f,1}(\boldsymbol{Z}, \boldsymbol{Z}) & \cdots & k_{d_f,d_f}(\boldsymbol{Z}, \boldsymbol{Z}) \end{bmatrix} \quad \boldsymbol{K}(\boldsymbol{Z}, \boldsymbol{z}) = \begin{bmatrix} k_{1,1}(\boldsymbol{Z}, \boldsymbol{z}) & \cdots & k_{1,d_f}(\boldsymbol{Z}, \boldsymbol{z}) \\ \vdots & \ddots & \vdots \\ k_{d_f,1}(\boldsymbol{Z}, \boldsymbol{z}) & \cdots & k_{d_f,d_f}(\boldsymbol{Z}, \boldsymbol{z}) \end{bmatrix}$$

$$\hat{\boldsymbol{f}}(\boldsymbol{Z}) = \begin{bmatrix} \hat{f}_1(\boldsymbol{Z}) & \cdots & \hat{f}_{d_f}(\boldsymbol{Z}) \end{bmatrix}^T. \tag{12}$$

3.3. Output Decoupling through Linear Models of Coregionalization

While various positive definite kernels are known for scalar regression, positive definiteness is a major challenge in the multiple-output approach presented in Section 3.2 since it is not sufficient that each entry of $\boldsymbol{K}(\cdot, \cdot)$ is a covariance function. However, in the following we show that knowledge of the output correlation structure in the form of a matrix $\boldsymbol{A}$ allows to define proper kernel matrix functions via scalar covariance functions $k_i(\cdot, \cdot)$. For this, we require the following assumption.

Assumption 3 *Prior knowledge about the functions $f_i(\cdot)$ is expressed through independent prior GP distributions with scalar kernels $k_i(\cdot, \cdot)$, $i = 1, \dots, d_f$, i.e.,*

$$\boldsymbol{f}(\cdot) \sim \mathcal{GP}\left(\hat{\boldsymbol{f}}(\cdot), \operatorname{diag}\left(\begin{bmatrix} k_1(\cdot, \cdot) & \cdots & k_{d_f}(\cdot, \cdot) \end{bmatrix}\right)\right). \tag{13}$$

This assumption is not restrictive, since correlation in the training targets can be modeled through the matrix $\boldsymbol{A}$, and it is frequently used in the case where $\boldsymbol{A} = \boldsymbol{I}_{d_x}$ holds (Berkenkamp and Schoellig, 2015; Koller et al., 2018; Beckers et al., 2019; Hewing et al., 2020a).

Due to Assumption 3, it directly follows that

$$\boldsymbol{g}(\cdot) \sim \mathcal{GP}\left(\boldsymbol{A}\hat{\boldsymbol{f}}(\cdot), \boldsymbol{A}\mathrm{diag}\left(\begin{bmatrix} k_1(\cdot,\cdot) & \cdots & k_{d_f}(\cdot,\cdot) \end{bmatrix}\right)\boldsymbol{A}^T\right), \tag{14}$$

such that we can intuitively define a kernel matrix function through

$$\boldsymbol{K}(\boldsymbol{z},\boldsymbol{z}') = \boldsymbol{A}\mathrm{diag}\left(\begin{bmatrix} k_1(\boldsymbol{z},\boldsymbol{z}') & \cdots & k_{d_f}(\boldsymbol{z},\boldsymbol{z}') \end{bmatrix}\right)\boldsymbol{A}^T. \tag{15}$$

It is trivial to show that this kernel parameterization is a special case of a linear model of coregionalization (Álvarez et al., 2011), such that we can immediately extend the approach in Duvenaud (2014) to recover models for the individual functions $f_i(\cdot)$, as shown in the following lemma[2].

Lemma 1 *Consider a nonlinear system (1) with matrix* $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a}_1 & \cdots & \boldsymbol{a}_{d_f} \end{bmatrix}$ *composed of column vectors* $\boldsymbol{a}_i$*, for which a training data set* $\mathbb{D}_N$ *and prior distributions that satisfy Assumptions 1 and 3, respectively, are given. Then, the posterior distributions are given by*

$$f_i(\boldsymbol{z})|\mathbb{D} \sim \mathcal{N}(\mu_i(\boldsymbol{z}), \sigma_i(\boldsymbol{z})), \tag{16}$$

where

$$\mu_i(\boldsymbol{z}) = \hat{f}_i(\boldsymbol{z}) + \left(k_i^T(\boldsymbol{Z},\boldsymbol{z}) \otimes \boldsymbol{a}_i^T\right)\left(\boldsymbol{K}(\boldsymbol{Z},\boldsymbol{Z}) + \boldsymbol{\Sigma}_{\mathrm{on}} \otimes \boldsymbol{I}_N\right)^{-1}\left(\boldsymbol{t} - \hat{\boldsymbol{f}}(\boldsymbol{Z})\right) \tag{17}$$

$$\sigma_i^2(\boldsymbol{z}) = k_i(\boldsymbol{z},\boldsymbol{z}) - \left(k_i^T(\boldsymbol{Z},\boldsymbol{z}) \otimes \boldsymbol{a}_i^T\right)\left(\boldsymbol{K}(\boldsymbol{Z},\boldsymbol{Z}) + \boldsymbol{\Sigma}_{\mathrm{on}} \otimes \boldsymbol{I}_N\right)^{-1}\left(k_i(\boldsymbol{Z},\boldsymbol{z}) \otimes \boldsymbol{a}_i\right). \tag{18}$$

A crucial benefit of this decoupling of the outputs is that it allows the application of scalar analysis methods to uniformly bound the regression error as proposed in Lederer et al. (2019a). This is formalized in the following theorem.

Theorem 2 *Consider a nonlinear system (1), a training data set* $\mathbb{D}_N$*, and prior distributions satisfying Assumptions 1-3, respectively. For any* $\delta \in (0,1)$*,* $\tau \in \mathbb{R}_+$*, and* $i = 1, \ldots, d_f$*, it holds that*

$$P\left(|f_i(\boldsymbol{z}) - \mu_i(\boldsymbol{z})| \leq \sqrt{\beta(N,\tau)}\sigma_i(\boldsymbol{z}) + \gamma_i(\tau) \quad \forall \boldsymbol{z} \in \mathbb{X} \times \mathbb{U}\right) \geq 1 - \delta, \tag{19}$$

where

$$\beta(\tau) = 2d_x \log\left(1 + \frac{r_0}{\tau}\right) - \log(\delta) \tag{20}$$

$$\gamma_i(\tau) = (L_{\mu_i} + L_{f_i})\tau + \sqrt{\beta(\tau) L_{\sigma_i^2}\tau}. \tag{21}$$

Here, L_{μ_i} *and* $L_{\sigma_i^2}$ *are the Lipschitz constants of the mean and variance functions, respectively, and* $r_0 = \max_{\boldsymbol{z},\boldsymbol{z}' \in \mathbb{X}\times\mathbb{U}} \|\boldsymbol{z} - \boldsymbol{z}'\|$.

This theorem is a generalization of Lederer et al. (2021, Lemma 1) and many properties directly transfer. Small error bounds can be achieved through small posterior standard deviations $\sigma_i(\boldsymbol{z})$, which corresponds to high data densities. This resembles well-known relationships from scattered data approximation (Wendland, 2004) and Bayesian optimization (Srinivas et al., 2012). The dependence of the uniform error bound (19) on the constants $\gamma_i(\tau)$ does not affect this behavior, since they can be chosen arbitrarily small, and convergence to 0 can be shown under weak assumptions on $\sigma_i(\boldsymbol{z})$ (Lederer et al., 2019a).

Remark 3 *Theorem 2 admits the unintuitive behavior that adding training samples can lead to a locally higher uniform error bound. This is due to the fact that adding data in some regions can increase the Lipschitz constants of* $\mu(\cdot)$ *and* $\gamma(\cdot)$ *(Lederer et al., 2019a), and thereby increase the uniform error bound in other regions. Note that a similar argument can be used for uniform error bounds based on RKHS theory (Srinivas et al., 2012; Chowdhury and Gopalan, 2017).*

2. Proofs for all theoretical results can be found in the appendix.

4. Control-Based Information Measures

While the uniform error bound in Theorem 2 establishes a connection between the training data distribution, represented by the posterior GP variance, and the regression performance, it is ignorant of the control task. In order to measure the importance of data for control performance, we consider the Lyapunov stability conditions (Khalil, 2002) for the closed loop system, which require a negative derivative of the Lyapunov function $V(\cdot,\cdot)$, i.e.,

$$\dot{V}(\boldsymbol{x},t) = (\nabla V(\boldsymbol{x}))^T \boldsymbol{A}\tilde{\boldsymbol{f}}(\boldsymbol{x}) + \frac{\partial}{\partial t}V(\boldsymbol{x},t), \tag{22}$$

where we extend the shorthand notation for the closed loop introduced in (3) to $\tilde{\boldsymbol{f}}(\cdot)$, the GP mean $\tilde{\boldsymbol{\mu}}(\cdot)$ and the GP variance $\tilde{\boldsymbol{\sigma}}^2(\cdot)$. Although the function $\tilde{\boldsymbol{f}}(\cdot)$ is unknown, we can bound the Lyapunov function derivative based on the uniform error bound (19), which yields

$$\dot{V}(\boldsymbol{x},t) \leq \dot{V}_{\mathrm{nom}}(\boldsymbol{x},t) + \dot{V}_{\boldsymbol{\sigma}}(\boldsymbol{x},t), \tag{23}$$

where we decouple the bound into the nominal and the uncertain component

$$\dot{V}_{\mathrm{nom}}(\boldsymbol{x},t) = (\nabla V(\boldsymbol{x},t))^T \boldsymbol{A}\tilde{\boldsymbol{\mu}}(\boldsymbol{x}) + \frac{\partial}{\partial t}V(\boldsymbol{x},t) \tag{24}$$

$$\dot{V}_{\boldsymbol{\sigma}}(\boldsymbol{x},t) = \begin{bmatrix} \sum_{i=1}^{d_x} \left| a_{i,1} \frac{\partial}{\partial x_i} V(\boldsymbol{x},t) \right| & \cdots & \sum_{i=1}^{d_x} \left| a_{i,d_f} \frac{\partial}{\partial x_i} V(\boldsymbol{x},t) \right| \end{bmatrix} \left(\sqrt{\beta(\tau)}\tilde{\boldsymbol{\sigma}}(\boldsymbol{x}) + \boldsymbol{\gamma}(\tau) \right), \tag{25}$$

respectively, with $\boldsymbol{\gamma}(\tau) = \begin{bmatrix} \gamma_1(\tau) & \cdots & \gamma_{d_f}(\tau) \end{bmatrix}^T$. Since the posterior mean function $\tilde{\boldsymbol{\mu}}(\cdot)$ is known, negative definiteness of the nominal Lyapunov function derivative $\dot{V}_{\mathrm{nom}}(\cdot,\cdot)$ can typically be ensured through a suitable control law $\boldsymbol{\pi}(\cdot, \boldsymbol{x}^{(1)}, \boldsymbol{u}^{(1)}, \ldots, t)$, and we assume this in the following. As a result, stability of the closed-loop system depends on the magnitude of the uncertain Lyapunov function derivative $\dot{V}_{\boldsymbol{\sigma}}(\cdot,\cdot)$, which is strongly influenced by posterior GP standard deviations $\tilde{\boldsymbol{\sigma}}(\cdot)$. Although this establishes a direct relationship between the training data density and the control task, the dependency of $\tilde{\boldsymbol{\sigma}}(\cdot)$ on training samples is highly nonlinear and the computation of $\tilde{\boldsymbol{\sigma}}(\cdot)$ is computationally expensive. In order to mitigate these issues, we introduce the weighted M-fill distances, in analogy to the M-fill distance proposed in Lederer et al. (2021).

Definition 4 *The weighted M-fill distances $\phi_i(\boldsymbol{x}, \mathbb{D}_N)$ at a point $\boldsymbol{x}$ are defined as the minimum radii φ of balls with center $[\boldsymbol{x}^T \ \boldsymbol{\pi}^T(\boldsymbol{x})]^T$, such that the balls contain M samples $\boldsymbol{z}^{(n)}$, i.e.,*

$$\tilde{\phi}_i(\boldsymbol{x}, \mathbb{D}_N) = \min_{\phi \in \mathbb{R}_{+,0}} \varphi \tag{26a}$$

$$\text{such that} \left| \left\{ \boldsymbol{z}^{(n)} \in \mathbb{D}_N : \sqrt{\sum_{j=1}^{d_x} \frac{\left(x_j - x_j^{(n)}\right)^2}{l_{i,j}^2} + \sum_{j=1}^{d_u} \frac{\left(\pi_j(\boldsymbol{x}) - u_j^{(n)}\right)^2}{l_{i,d_x+j}^2}} \leq \varphi \right\} \right| \geq M, \tag{26b}$$

where $|\cdot|$ denotes the cardinality of the set and we use the abbreviation $\boldsymbol{\pi}(\boldsymbol{x}) = \boldsymbol{\pi}(\boldsymbol{x}, \boldsymbol{x}^{(1)}, \boldsymbol{u}^{(1)}, \ldots, t)$.

The weighted M-fill distances measure the distance from a test point $[\boldsymbol{x}^T \ \boldsymbol{\pi}^T(\boldsymbol{x})]^T$ to the M closest training samples in the Mahalonobis distance metric induced by the length scales $\boldsymbol{l}_i$ of the squared exponential kernels (4). By choosing a small number $M \ll N$, only training points in the proximity of the test point $[\boldsymbol{x}^T \ \boldsymbol{\pi}^T(\boldsymbol{x})]^T$ are relevant for the weighted M-fill distance $\tilde{\phi}_i(\boldsymbol{x}, \mathbb{D}_N)$. This allows us to measure the local data density in a flexible way, where high training data densities are indicated by low values of $\tilde{\phi}_i(\boldsymbol{x}, \mathbb{D}_N)$. Moreover, it is possible to bound the posterior GP variances $\tilde{\boldsymbol{\sigma}}^2(\boldsymbol{x})$ in

terms of the weighted M-fill distances $\tilde{\phi}_i(\boldsymbol{x}, \mathbb{D}_N)$[3]We exploit this property in the following theorem to derive conditions that guarantee that the summands of the uncertain Lyapunov derivative $\dot{V}_{\boldsymbol{\sigma}}(\cdot,\cdot)$ are upper bounded by functions $\xi_i : \mathbb{R}^{d_x+d_u} \times \mathbb{R}_{0,+} \to \mathbb{R}_{0,+}$, $i = 1, \ldots, d_f$. For suitably chosen functions $\xi_i(\cdot,\cdot)$, the satisfaction of these bounds implies stability of the closed-loop system.

Theorem 5 *Choose τ such that $\sqrt{\beta(\tau)}\tilde{\sigma}_i(\boldsymbol{x}) > \gamma_i(\tau)$ holds for all $\boldsymbol{x} \in \mathbb{X}$ and assume*

$$4\beta(\tau)s_{f_i}^2 \left(\sum_{j=1}^{d_x} \left| a_{j,i} \frac{\partial}{\partial x_j} V(\boldsymbol{x}, t) \right| \right)^2 > \xi_i^2(\boldsymbol{x}, t). \tag{27}$$

If the M-fill distance $\tilde{\phi}_i(\cdot, \mathbb{D}_N)$ satisfies $\tilde{\phi}_i^2(\boldsymbol{x}, \mathbb{D}_N) \leq \bar{\phi}_i^2(\boldsymbol{x}, t) + \theta_i^2$ for all $\boldsymbol{z} \in \mathbb{X} \times \mathbb{U}$, where

$$\bar{\phi}_i^2(\boldsymbol{x}, t) = -\log \left(1 - \frac{\xi_i^2(\boldsymbol{x}, t)}{4\beta(\tau)s_{f_i}^2 \left(\sum_{j=1}^{d_x} \left| a_{j,i} \frac{\partial}{\partial x_j} V(\boldsymbol{x}, t) \right| \right)^2} \right) \tag{28}$$

$$\theta_i^2 = \log \left(\frac{s_{f_i}^2 \sum_{j=1}^{d_x} a_{j,i}^2}{\max_{m=1,\ldots,d_x} \sum_{n=1}^{d_f} \sum_{j=1}^{d_x} |a_{m,n}a_{j,n}| s_{f_n}^2 + \frac{\lambda_{\max}(\Sigma_{\text{on}})}{M}} \right), \tag{29}$$

then, $\sum_{j=1}^{d_x} \left| a_{j,i} \frac{\partial}{\partial x_j} V(\boldsymbol{x}, t) \right| \left(\sqrt{\beta(\tau)}\tilde{\sigma}_i(\boldsymbol{x}) + \gamma_i(\tau) \right) \leq \xi_i(\boldsymbol{x}, t)$ holds for all $\boldsymbol{x} \in \mathbb{X}$.

Condition (27) is necessary to ensure the existence of the logarithm in (28), but it is not restrictive since $\xi_i^2(\cdot,\cdot)$ are upper bounds. Hence, we can simply tighten the bounds until $\xi_i^2(\cdot,\cdot)$ satisfies condition (27). The expressions (28) and (29) have different roles. The values θ_i express the difficulty of recovering the functions $f_i(\cdot)$ from the noisy measurements of $\boldsymbol{A}\boldsymbol{f}(\cdot)$, which in turn depends on the signal variances $s_{f_i}^2$ of the independent covariance functions $k_i(\cdot,\cdot)$ and the magnitude of the elements of $\boldsymbol{A}$. The larger the corresponding entries of $\boldsymbol{A}$ and the signal variance $s_{f_i}^2$ are, the easier it is to learn $f_i(\cdot)$, and this corresponds to a comparatively large value of θ_i^2. In contrast, $\bar{\phi}_i^2(\cdot,\cdot)$ captures the dependency on the control task. The numerator in (28) corresponds to a summand of the uncertain Lyapunov derivative component (25) without any training data. Therefore, $\bar{\phi}_i^2(\boldsymbol{x}, t)$ goes to ∞ when the prior uncertain Lyapunov component converges to the bound $\xi_i^2(\boldsymbol{x}, t)$, which intuitively represents the fact that few data samples are required in this case.

As mentioned previously, Theorem 5 can be used to analyze the stability of the closed loop system. More specifically, stability is guaranteed if the negated nominal Lyapunov derivative $\dot{V}_{\text{nom}}(\cdot,\cdot)$ is larger than the uncertain component $\dot{V}_{\boldsymbol{\sigma}}(\cdot,\cdot)$. This can be trivially checked with Theorem 5 by defining $\xi_j(\cdot,\cdot)$ such that $\sum_{i=1}^{d_f} \xi_i(\boldsymbol{x}, t) \leq |\dot{V}_{\text{nom}}(\boldsymbol{x}, t)|$. A natural choice satisfying this condition together with constraint (27) is given by

3. A bound for the posterior variance in terms of the M-fill distance is derived in the appendix.

$$\xi_i(\boldsymbol{x},t) = \min\left\{ -\frac{\dot{V}_{\text{nom}}(\boldsymbol{x},t)\sum_{j=1}^{d_x}|a_{j,i}|}{\sum_{j=1}^{d_x}\sum_{n=1}^{d_f}|a_{j,n}|}, 2\sqrt{\beta(\tau)}s_{f_i}\left|\sum_{j=1}^{d_x} a_{j,i}\frac{\partial}{\partial x_j}V(\boldsymbol{x},t)\right| - \nu \right\}, \tag{30}$$

where $\nu \in \mathbb{R}_+$ is an arbitrarily small constant. Furthermore, convergence rates can be examined in a similar way. For example, an exponential rate of convergence is achieved by guaranteeing that a requirement similar to (30) is satisfied, where $\dot{V}_{\text{nom}}(\boldsymbol{x},t)$ is replaced by $\dot{V}_{\text{nom}}(\boldsymbol{x},t) + V(\boldsymbol{x},t)$.

Due to the intuitive interpretation of Theorem 5, we propose to use it as the basis for a measure of the importance of training data for control. This naturally leads to the definition of the ρ-gap.

Definition 6 *The ρ-gap is defined as*

$$\rho(\boldsymbol{x},t,\mathbb{D}_N) = \sum_{i=1}^{d_f} \max\{0, \phi_i^2(\boldsymbol{x},\mathbb{D}_N) - \bar{\phi}_i^2(\boldsymbol{x},t) - \theta_i^2\}. \tag{31}$$

Essentially, the ρ-gap measures the discrepancy between the required data density, which is expressed through $\bar{\phi}_i^2(\boldsymbol{x},t) + \theta_i^2$ and depends on the desired bounds $\xi_i(\cdot,\cdot)$, the Lyapunov derivative and the signal standard deviations $s_{f_i}^2$, and the actual data density represented by the M-fill distances $\phi_i^2(\boldsymbol{x},\mathbb{D}_N)$, which are independent of the control problem and only depend on the available data.

5. Data Selection Strategies

Based on the information measure proposed in Section 4, the data set can be preprocessed to contain only the most relevant information for the given control task. This becomes particularly important in scenarios where the prediction of the GP model must be performed under tight real-time constraints. Even with a precomputation of the matrix inverse in (17) (which takes $\mathcal{O}(N^3)$), the N kernel evaluations (for $k_m^T(\boldsymbol{Z},\boldsymbol{z})$) and the corresponding multiplications ($\mathcal{O}(N)$ for the posterior mean, $\mathcal{O}(N^2)$ for the posterior variance) must still be performed online. In practice, this imposes an upper bound for the number of points that can be considered by the model. We formulate the resulting computational constraint independent of the hardware and specific real-time limit as follows.

Assumption 4 *The computational constraints allow a maximum of $\bar{N}$ data points to be considered by the GP regression model* (17).

For the non-trivial case $N > \bar{N}$, this makes a selection of an active data set $\mathbb{D}_{\bar{N}} \subset \mathbb{D}_N$ necessary. Such a data selection has been considered for general function learning in Krause et al. (2008), and specifically for control tasks in Umlauft et al. (2020). But both employ entropy-based criteria, which only aims to optimize the precision of the model but does not consider the closed-loop control performance. Therefore, we utilize the ρ-gap as the measure for the control performance of a data set to find the optimal active data set[4]

$$\beth^* = \underset{\beth\in\mathbb{P}^{\mathbb{I}}_{\bar{N}}}{\arg\min} \max_{t\in\mathbb{T},\boldsymbol{x}\in\mathbb{X}} \rho\left(\boldsymbol{x},t,\beth\right), \tag{32}$$

where $\mathbb{T} = [0,T[$ with initial time $t_0 \in \mathbb{R}_{0,+}$, $t_0 < \infty$ and (possibly infinite) final time $T \in \mathbb{R}\cup\infty$. The optimal active data set is then given by $\mathbb{D}_{\bar{N}} = \left\{\boldsymbol{z}^{(\beth(i))}, \boldsymbol{y}^{(\beth(i))}\right\}_{i=1}^{\bar{N}}$. If the desired trajectory

4. To simplify notation, we introduce the index set $\mathbb{I} = \{1,\dots,N\}$ and all possible subsets with size $\bar{N}$, denoted as $\mathbb{P}^{\mathbb{I}}_{\bar{N}}$. $\beth(i)$ denotes the i-th entry of the index set $\beth$.

Algorithm 1: Greedy optimization for optimal subset selection

Input: $\mathbb{D}_N$, $\rho(\cdot,\cdot)$, $\mathbb{T}_0,\dots,\mathbb{T}_S$
Output: $\mathbb{D}_{\bar{N}}^{\mathbb{T}_0},\dots,\mathbb{D}_{\bar{N}}^{\mathbb{T}_S}$
for $\mathbb{T}=\mathbb{T}_0,\dots,\mathbb{T}_S$ **do**
  $\mathbb{D}_{\bar{N}}^{\mathbb{T}}\leftarrow\emptyset$, $\mathbb{I}=\{1,\dots,N\}$
  for $n=0,\dots,\bar{N}$ **do**
    $i^*,t^*\leftarrow\underset{i\in\mathbb{I},t\in\mathbb{T}}{\arg\max}\,\rho\left(\boldsymbol{x}^{(i)},t,\mathbb{D}_{\bar{N}}^{\mathbb{T}}\right)$
    $\mathbb{D}_{\bar{N}}^{\mathbb{T}}\leftarrow\mathbb{D}_{\bar{N}}^{\mathbb{T}}\cup\left\{\boldsymbol{z}^{(i^*)},\boldsymbol{y}^{(i^*)}\right\}$
    $\mathbb{I}\leftarrow\mathbb{I}\setminus\{i^*\}$
  end
end

has a wide spread or $\bar{N}$ is small, then the selected subset might not lead to a satisfactory control performance. For such a case, we can partition the task in $S\in\mathbb{N}$ time intervals $\mathbb{T}_0=[0,t_1[$, $\mathbb{T}_1=[t_1,t_2[,\dots,\mathbb{T}_S=[t_S,T[$ and compute the corresponding optimal subsets $\mathbb{D}_{\bar{N}}^{\mathbb{T}_0},\dots,\mathbb{D}_{\bar{N}}^{\mathbb{T}_S}$.

Due to its mixed nature (combinatorial in $\sqsupset$, continuous in t and $\boldsymbol{x}$), the optimization problem (32) is not trivial to solve. However, the optimization can be performed offline, assuming that sufficient memory capacity is available to store all precomputed subsets. Furthermore, in the field of function learning it has been shown that greedy algorithms can show near-optimal behavior (Krause et al., 2008). Therefore, we propose the greedy data selection procedure shown in Algorithm 1.

6. Numerical Evaluation

In order to evaluate the proposed importance measure, we consider the nonlinear system

$$\dot{\boldsymbol{x}}=\boldsymbol{x}+\frac{1}{1+\exp(-2x_1)}\begin{bmatrix}1\\-1\end{bmatrix}+0.5\begin{bmatrix}\sin(\pi x_2)\\\cos(\pi x_1)\end{bmatrix}+\boldsymbol{u}, \tag{33}$$

which is a slight modification of the example proposed in (Umlauft et al., 2018). We assume a prior model $\hat{\boldsymbol{f}}(\boldsymbol{z})=\boldsymbol{x}+\boldsymbol{u}$, and define the kernel matrix using

$$\boldsymbol{A}=\begin{bmatrix}1&0\\-1&1\end{bmatrix} \tag{34}$$

and squared exponential kernels $k_1(\boldsymbol{x},\boldsymbol{x}')$, $k_2(x_1,x_1')$. This ensures that the correlation between the outputs caused by the second summand in (33) is properly modeled. Furthermore, we employ a control law

$$\boldsymbol{\pi}(\boldsymbol{x},t)=-(\boldsymbol{\mu}(\boldsymbol{x})+K(\boldsymbol{x}-\boldsymbol{x}_{\mathrm{ref}}(t))-\dot{\boldsymbol{x}}_{\mathrm{ref}}(t)), \tag{35}$$

with gain $K=15$ and reference trajectories $\boldsymbol{x}_{\mathrm{ref}}(t)=[c_1\sin(t)\ c_2\cos(t)]^T$ with randomly drawn $c_i\sim\mathcal{N}(0,1)$. Nominal stability of the closed loop is trivially shown using the Lyapunov function $V(\boldsymbol{x},t)=(\boldsymbol{x}-\boldsymbol{x}_{\mathrm{ref}}(t))^T(\boldsymbol{x}-\boldsymbol{x}_{\mathrm{ref}}(t))$. The training set is generated by simulating the closed-loop system with prior mean $\boldsymbol{\mu}(\cdot)=\hat{\boldsymbol{f}}(\cdot)$ and sampling $N=100$ data points during the interval $t\in[0,T]$ with simulation time $T=10$. We divide the period of the reference trajectory into $S=10$ equally long time intervals $\mathbb{T}_s$ and select subsets of cardinality $\bar{N}=10$ for each interval using Algorithm 1 with $M=1$ and $\xi_i(\cdot)$ as defined in (30).

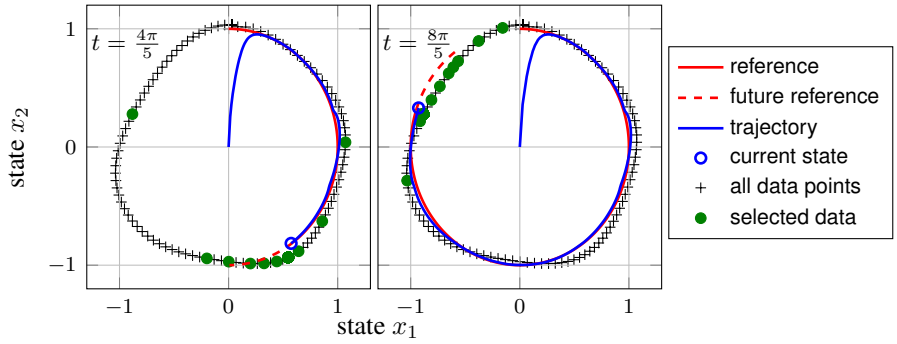

Figure 1: The desired and actual trajectory of the dynamical system. Training data selected by Algorithm 1 lies close to the future reference and ensures a high tracking accuracy.

Snapshots of the selected subsets and the resulting system trajectories are illustrated in Fig. 1. It can be clearly seen that the training samples are generally chosen close to the reference in the considered time intervals $\mathbb{T}_s$. This is because the feedback $K(\boldsymbol{x} - \boldsymbol{x}_{\mathrm{ref}})$ ensures that the Lyapunov stability condition is satisfied far away from the reference regardless of training data. Moreover, the data density increases as the distance to the reference decreases due to the vanishing effect of the feedback in its proximity.

We evaluate the proposed technique by carrying out 100 control law roll-outs with randomly drawn trajectory parameters c_i. Furthermore, we compare the results with the performance of the full data set, as well as data selected with a greedy maximization of the mutual information with respect to a uniform grid over $[-1.5, 1.5]^2$, and maximization of the mutual information with respect to the considered trajectory interval (Umlauft and Hirche, 2020). The results are depicted in Table 1. A reduction in the average computation time by a factor of approximately 10 for all subset selection methods can observed, which is a straightforward consequence of the linear complexity of predictions (Rasmussen and Williams, 2006). Moreover, the steady state mean squared tracking error is the lowest for the subset selected based on the ρ-gap, and even smaller than for the full data set. The approach from (Umlauft and Hirche, 2020) exhibits a similar but slightly worse performance. Although it might seem unintuitive that fewer training samples lead to a better control performance, this is theoretically covered by Theorem 2 as discussed in Remark 3. This fact underlines the importance of selecting relevant training samples for learning-based control.

7. Conclusion

We presented the ρ-gap, a measure that quantifies the value of data for a broad class of control tasks. The proposed quantity is used to identify the optimal data set for control tasks under computational constraints. Simulations demonstrate that the data subsets selected using the presented measure are highly correlated with the control task and can even be beneficial for control performance.

Table 1: Tracking errors and GP prediction times resulting from different subset selection criteria. The ρ-gap significantly outperforms existing methods regarding control performance.

criterion	full data set	mutual information w.r.t. uniform grid	mutual information w.r.t. reference	ρ-gap
steady-state MSE ($\cdot 10^{-3}$)	1.15	1.32	0.38	**0.16**
prediction time (μs)	437	**45.0**	**45.0**	**45.0**

References

Tansu Alpcan and Iman Shames. An Information-Based Learning Approach to Dual Control. *IEEE Transactions on Neural Networks and Learning Systems*, 26(11):2736–2748, 2015. ISSN 21622388. doi: 10.1109/TNNLS.2015.2392122.

Mauricio A. Álvarez, Lorenzo Rosasco, and Neil D. Lawrence. Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning*, 4(3):195–266, 2011. ISSN 19358237. doi: 10.1561/2200000036.

Anil Aswani, Humberto Gonzalez, S. Shankar Sastry, and Claire Tomlin. Provably safe and robust learning-based model predictive control. *Automatica*, 49(5):1216–1226, 2013. ISSN 00051098. doi: 10.1016/j.automatica.2013.02.003.

Thomas Beckers, Dana Kulić, and Sandra Hirche. Stable Gaussian Process based Tracking Control of Euler-Lagrange Systems. *Automatica*, 103(23):390–397, 2019.

Felix Berkenkamp and Angela P. Schoellig. Safe and Robust Learning Control with Gaussian Processes. In *Proceedings of the European Control Conference*, pages 2496–2501, 2015.

Alexandre Capone and Sandra Hirche. Backstepping for Partially Unknown Nonlinear Systems Using Gaussian Processes. *IEEE Control Systems Letters*, 3(2):416–421, 2019.

Alexandre Capone, Gerrit Noske, Jonas Umlauft, Thomas Beckers, Armin Lederer, and Sandra Hirche. Localized active learning of Gaussian process state space models. In *Learning for Dynamics & Control*, pages 1–10, 2020.

Alexandre Capone, Armin Lederer, Jonas Umlauft, and Sandra Hirche. Data Selection for Multi-Task Learning under Dynamic Constraints. *IEEE Control Systems Letters*, 5(3):959–964, 2021. ISSN 24751456. doi: 10.1109/LCSYS.2020.3006279.

Sayak Ray Chowdhury and Aditya Gopalan. On Kernelized Multi-armed Bandits. In *Proceedings of the International Conference on Machine Learning*, pages 844–853, 2017.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep Reinforcement Learning in a Handful of Trials using Probabilistic Dynamics Models. In *Advances in Neural Information Processing Systems*, pages 4754–4765, 2018.

Marc Peter Deisenroth, Dieter Fox, and Carl Edward Rasmussen. Gaussian Processes for Data-Efficient Learning in Robotics and Control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):408–423, 2015. ISSN 01628828. doi: 10.1109/TPAMI.2013.218.

David Kristjanson Duvenaud. *Automatic Model Construction with Gaussian Processes*. PhD thesis, University of Cambridge, 2014.

Jaime F. Fisac, Anayo K. Akametalu, Melanie N. Zeilinger, Shahab Kaynama, Jeremy Gillula, and Claire J. Tomlin. A General Safety Framework for Learning-Based Control in Uncertain Robotic Systems. *IEEE Transactions on Automatic Control*, 64(7):2737–2752, 2019. ISSN 15582523.

Aditya Gahlawat, Pan Zhao, Andrew Patterson, Naira Hovakimyan, and Evangelos A. Theodorou. $\mathcal{RL}_1$-$\mathcal{GP}$: $\mathcal{L}_1$ Adaptive Control with Bayesian Learning. In *Learning for Dynamics & Control*, pages 1–15, 2020. URL http://arxiv.org/abs/2004.14594.

Lukas Hewing, Elena Arcari, Lukas P. Fröhlich, and Melanie N. Zeilinger. On simulation and trajectory prediction with Gaussian process dynamics. In *Learning for Dynamics & Control*, pages 1–11, 2020a.

Lukas Hewing, Juraj Kabzan, and Melanie N. Zeilinger. Cautious Model Predictive Control using Gaussian Process Regression. *IEEE Transactions on Control Systems Technology*, 28(6):2736–2743, 2020b.

Hassan K. Khalil. *Nonlinear Systems*. Prentice-Hall, Upper Saddle River, NJ, third edition, 2002.

Torsten Koller, Felix Berkenkamp, Matteo Turchetta, and Andreas Krause. Learning-based Model Predictive Control for Safe Exploration. In *Proceedings of the IEEE Conference on Decision and Control*, pages 6059–6066, 2018.

A Krause, Aarti Singh, and C Guestrin. Near-optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies. *Journal of Machine Learning Research*, 9: 235–284, 2008.

Armin Lederer, Jonas Umlauft, and Sandra Hirche. Uniform Error Bounds for Gaussian Process Regression with Application to Safe Control. In *Advances in Neural Information Processing Systems*, 2019a.

Armin Lederer, Jonas Umlauft, and Sandra Hirche. Posterior Variance Analysis of Gaussian Processes with Application to Average Learning Curves. 2019b. URL http://arxiv.org/abs/1906.01404.

Armin Lederer, Alexandre Capone, and Sandra Hirche. Parameter Optimization for Learning-based Control of Control-Affine Systems. In *Learning for Dynamics and Control*, volume 120, pages 1–11, 2020.

Armin Lederer, Alexandre Capone, Jonas Umlauft, and Sandra Hirche. How Training Data Impacts Performance in Learning-based Control. *IEEE Control Systems Letters*, 5(3):905–910, 2021. URL http://arxiv.org/abs/2005.12062.

Friedrich Pukelsheim. *Optimal Design of Experiments*. Society for Industrial and Applied Mathematics, 2006. doi: 10.1137/1.9780898719109. URL https://epubs.siam.org/doi/abs/10.1137/1.9780898719109.

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

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012.

Jonas Umlauft and Sandra Hirche. Feedback Linearization based on Gaussian Processes with event-triggered Online Learning. *IEEE Transactions on Automatic Control*, 2020.

Jonas Umlauft, Lukas Pöhler, and Sandra Hirche. An Uncertainty-Based Control Lyapunov Approach for Control-Affine Systems Modeled by Gaussian Process. *IEEE Control Systems Letters*, 2(3):483–488, 2018.

Jonas Umlauft, Thomas Beckers, Alexandre Capone, Armin Lederer, and Sandra Hirche. Smart Forgetting for Safe Online Learning with Gaussian Processes. In *Learning for Dynamics & Control*, pages 1–10, 2020.

Holger Wendland. *Scattered Data Approximation*. Cambridge University Press, 2004.

Appendix A. Component-wise Uniform Error Bounds for Multiple-Output Gaussian Process Regression

Proof of Lemma 1 It can be easily checked that we can express the kernel matrix function as

$$\boldsymbol{K}(\boldsymbol{z}, \boldsymbol{z}') = \sum_{i=1}^{d_f} \boldsymbol{B}_i k_i(\boldsymbol{z}, \boldsymbol{z}'), \tag{36}$$

where

$$\boldsymbol{B}_i = \begin{bmatrix} a_{1,i} \\ \vdots \\ a_{d_x,i} \end{bmatrix} \underbrace{\begin{bmatrix} a_{1,i} & \cdots & a_{d_x,i} \end{bmatrix}}_{\boldsymbol{a}_i^T}. \tag{37}$$

Therefore, the kernel matrix is linear in the scalar kernel functions $k_m(\cdot,\cdot)$, such that the posterior of $\boldsymbol{a}_i f_i(\cdot)$ can be obtained as

$$\boldsymbol{a}_i f_i(\cdot) | \mathbb{D}_N \sim \mathcal{N}(\mu_{\boldsymbol{a}_i f_i}(\cdot), \boldsymbol{\Sigma}_{\boldsymbol{a}_i f_i}(\cdot)), \tag{38}$$

where

$$\mu_{\boldsymbol{a}_i f_i}(\boldsymbol{z}) = \left(\boldsymbol{B}_i \otimes k_i^T(\boldsymbol{Z}, \boldsymbol{z})\right) \left(\boldsymbol{\Sigma}_{\text{on}} \otimes \boldsymbol{I}_N + \sum_{j=1}^{d_f} \boldsymbol{B}_j \otimes k_j(\boldsymbol{Z}, \boldsymbol{Z}) \right)^{-1} \left(\boldsymbol{t} - \hat{\boldsymbol{f}}(\boldsymbol{Z}) \right) \tag{39}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{a}_i f_i}(\boldsymbol{z}) = k_i(\boldsymbol{z}, \boldsymbol{z}) \boldsymbol{B}_i - \left(\boldsymbol{B}_i \otimes k_i^T(\boldsymbol{Z}, \boldsymbol{z})\right) \left(\boldsymbol{\Sigma}_{\text{on}} \otimes \boldsymbol{I}_N + \sum_{j=1}^{d_f} \boldsymbol{B}_j \otimes k_j(\boldsymbol{Z}, \boldsymbol{Z}) \right)^{-1} \left(\boldsymbol{B}_i \otimes k_i(\boldsymbol{Z}, \boldsymbol{z})\right) \tag{40}$$

This follows from a trivial extension of the results in (Duvenaud, 2014) to multiple-output GPs. Due to the definition of $\boldsymbol{B}_i$, we can equivalently write

$$\mu_{\boldsymbol{a}_i f_i}(\boldsymbol{z}) = \boldsymbol{a}_i \left(\left(\boldsymbol{a}_i^T \otimes k_i^T(\boldsymbol{Z}, \boldsymbol{z})\right) \left(\boldsymbol{\Sigma}_{\text{on}} \otimes \boldsymbol{I}_N + \sum_{j=1}^{d_f} \boldsymbol{a}_j \boldsymbol{a}_j^T \otimes k_j(\boldsymbol{Z}, \boldsymbol{Z}) \right)^{-1} \left(\boldsymbol{t} - \hat{\boldsymbol{f}}(\boldsymbol{Z}) \right) \right) \tag{41}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{a}_i f_i}(\boldsymbol{z}) = \boldsymbol{a}_i k_i(\boldsymbol{z}, \boldsymbol{z}) \boldsymbol{a}_m^T \tag{42}$$

$$- \boldsymbol{a}_i \left(\left(\boldsymbol{a}_i^T \otimes k_i^T(\boldsymbol{Z}, \boldsymbol{z})\right) \left(\boldsymbol{\Sigma}_{\text{on}} \otimes \boldsymbol{I}_N + \sum_{j=1}^{d_f} \boldsymbol{a}_j \boldsymbol{a}_j^T \otimes k_j(\boldsymbol{Z}, \boldsymbol{Z}) \right)^{-1} \left(\boldsymbol{a}_i \otimes k_i(\boldsymbol{Z}, \boldsymbol{z}) \right) \right) \boldsymbol{a}_i^T, \tag{43}$$

from which we can directly deduce the identities (17) and (18). ■

Proof of Theorem 2 The result follows from Lemma 1 and a straightforward adaption of (Lederer et al., 2019a, Theorem 3.1). ■

Appendix B. Variance Bounds and Lyapunov-Based Data Densities

Lemma 7 *The posterior variance $\tilde{\sigma}_i^2(\boldsymbol{x})$ defined in (18) is bounded by*

$$\tilde{\sigma}_j^2(\boldsymbol{x}) \leq s_{f_j}^2 - \frac{s_{f_j}^4 \exp(-\tilde{\phi}_j^2(\boldsymbol{x})) \sum_{i=1}^{d_x} a_{i,j}^2}{\max\limits_{m=1,\dots,d_x} \sum\limits_{n=1}^{d_f} \sum\limits_{i=1}^{d_x} a_{m,n} a_{i,n} s_{f_n}^2 + \frac{\lambda_{\max}(\Sigma_{\text{on}})}{M}}. \tag{44}$$

Proof This result is a direct extension of (Lederer et al., 2019b, Corollary 3.1) to multiple-output GPs with linear coregionalization and we pursue the proof analogously. Since the posterior variance is non-increasing, we can consider only training samples $\boldsymbol{z}^{(n)}$ within distance at most $\tilde{\phi}_j(\boldsymbol{x})$ to $\begin{bmatrix} \boldsymbol{x}^T & \boldsymbol{\pi}^T(\boldsymbol{x}) \end{bmatrix}^T$ in the posterior variance calculation. Therefore, we obtain

$$\sigma_j^2(\boldsymbol{x}) \leq s_{f_j}^2 - \frac{\| k_j^T(\boldsymbol{Z}_{\tilde{\phi}_j(\boldsymbol{x})}, \boldsymbol{z}) \otimes \boldsymbol{a}_j^T \|}{\lambda_{\max}\left(\boldsymbol{\Sigma}_{\text{on}} \otimes \boldsymbol{I}_N + \sum_{n=1}^{d_f} \boldsymbol{a}_n \boldsymbol{a}_n^T \otimes k_n(\boldsymbol{Z}_{\tilde{\phi}_j(\boldsymbol{x})}, \boldsymbol{Z}_{\tilde{\phi}_j(\boldsymbol{x})}) \right)}, \tag{45}$$

where $\boldsymbol{Z}_{\tilde{\phi}_j(\boldsymbol{x})}$ denotes the training samples with distance at most $\tilde{\phi}_j(\boldsymbol{x})$ to $\begin{bmatrix} \boldsymbol{x}^T & \boldsymbol{\pi}^T(\boldsymbol{x}) \end{bmatrix}^T$. We trivially obtain the bound

$$\| k_j(\boldsymbol{Z}_{\tilde{\phi}_j(\boldsymbol{x})}, \boldsymbol{z}) \otimes \boldsymbol{a}_j \| \geq M s_{f_j}^4 \exp(-\tilde{\phi}_j^2(\boldsymbol{x})) \sum_{i=1}^{d_x} a_{i,j}^2 \tag{46}$$

due to the distance restriction. Moreover, the application of Gershgorin's theorem yields

$$\lambda_{\max}\left(\boldsymbol{\Sigma}_{\text{on}} \otimes \boldsymbol{I}_N + \sum_{n=1}^{d_f} \boldsymbol{a}_n \boldsymbol{a}_n^T \otimes k_n(\boldsymbol{Z}_{\tilde{\phi}_j(\boldsymbol{x})}, \boldsymbol{Z}_{\tilde{\phi}_j(\boldsymbol{x})}) \right) \leq \lambda_{\max}(\boldsymbol{\Sigma}_{\text{on}}) + \max_{m=1,\dots,d_x} \sum_{n=1}^{d_f} \sum_{i=1}^{d_x} a_{m,n} a_{i,n} M s_{f_n}^2 \tag{47}$$

due to the definition of the M-fill distance $\tilde{\phi}_j(\boldsymbol{x})$ in Theorem 4. Substituting the bounds (46) and (47) in (45) finally yields the result. ■

Proof of Theorem 5 Since $\sqrt{\beta(\tau)}\tilde{\sigma}_j(\boldsymbol{x}) > \gamma_j(\tau)$ by assumption, we can simplify

$$\sqrt{\beta(\tau)}\tilde{\sigma}_j(\boldsymbol{x}) + \gamma_j(\tau) \leq 2\sqrt{\beta(\tau)}\tilde{\sigma}_j(\boldsymbol{x}). \tag{48}$$

Therefore, satisfaction of the condition

$$4\left(\sum_{i=1}^{d_x} |a_{i,j}\frac{\partial}{\partial x_i}V(\boldsymbol{x},t)|\right)^2 \beta(\tau)\tilde{\sigma}_j^2(\boldsymbol{x}) \leq \xi_j^2(\boldsymbol{x},t) \tag{49}$$

implies the statement of Theorem 5. Hence, we can substitute (44) and solve for $\tilde{\phi}_j^2(\boldsymbol{x})$ in order to prove Theorem 5. ■