

Learning Stochastically Stable Gaussian Process State-Space Models

Jonas Umlauf^{a,*}, Sandra Hirche^a

^a*Chair of Information-oriented Control, Department of Electrical and Computer Engineering, Technical University of Munich, Arcisstrasse 21, 80333 Munich, Germany*

Abstract

Control systems are increasingly applied in domains where an analytic description of the system dynamics does not exist or is difficult to obtain. Example applications include autonomous robots in unstructured environments, human behavior modeling for prediction and action recognition in human-machine-interaction, and chemical process industry. In many of these cases, classical system identification is challenging, because a parametric model structure is unknown. Data-driven nonparametric models such as Gaussian process state-space models (GPSSMs) offer a suitable alternative: GPSSMs are known for their data-efficiency and rely on Bayesian principles to include prior knowledge. However, properties like stability or boundedness are often known a priori, but rarely exploited during modeling. We therefore propose a novel approach for learning GPSSMs subject to stability constraints. Our approach enforces the convergence using control Lyapunov functions which are also obtained in a data-driven fashion. We analyze the resulting dynamics with respect to convergence radius and data collection. In simulation, we illustrate the precision of the identified model on a real-world dataset of goal-directed motions.¹

Keywords: System identification; Probabilistic models; State-space models; Gaussian processes; Machine learning; Stochastic modeling

*Corresponding author

Email addresses: jonas.umlaft@tum.de (Jonas Umlauf), hirche@tum.de (Sandra Hirche)

¹This is a preprint. The published article is available at <https://doi.org/10.1016/j.ifacsc.2020.100079>

1. INTRODUCTION

Model-based control techniques gain further importance as computational power is constantly becoming cheaper, not just in classical model predictive control schemes, but also in model-based reinforcement learning [51]. However, as the complexity of the dynamical systems is also becoming more evolved, the techniques known from classical system identification are often not sufficient to obtain precise models. For example, for control system applications including human behavior (such as social networks, human-robot interaction), it is evident, that parametric models may reach their limit because first order principles cannot be applied and the model structure is largely unknown. In general, for highly complex systems without analytic description, model structures with finite expressiveness are prone to fail. Fortunately, the collection and processing of data becomes also faster and cheaper and therefore data-driven models, which are considered to have infinite expressive power, are on the rise.

These data-driven models allow for very high flexibility, but are not necessarily coherent with physical principles, because no model structure is assumed. For example, energy dissipation is a fundamental property which holds for many physical system, but data-driven models do not necessarily adhere to this property. The dissipation of energy is closely entangled with stability of a dynamical system. Other examples are physical conservation laws [3] or convergence towards to fix point in case of modeling human goal-directed motions. In order to guarantee consistency of a model with the corresponding real system, it is essential to enforce these known properties into data-driven models. Incorporating this high-level prior knowledge is not only important for precise modeling but also absolutely necessary as without making prior assumptions, a generalization outside of the training points can not be expected as known from the *no-free-lunch* theorems [63].

This article will specifically focus on Gaussian processes (GPs) as underlying modeling paradigm. It is a kernel based method which allows to set the

smoothness characteristics of the model by a suitable kernel choice. It also handles noisy data intrinsically due to its stochastic nature and implicitly trades off bias and variance [44]. Most prominent advantage over other machine learning based identification techniques, e.g. neural networks, is its capability to quantify the model fidelity in a systematic way [24]. Additionally, its Bayesian background allows to inject prior knowledge based on a strong mathematical foundation. However, there exists no general approach how prior knowledge of the convergence behavior of a dynamical system is taken into account for a GPSSM. This is essential to ensure the model shows the same fundamental properties as the true system.

1.1. Related work

Classical system identification relies on a parametric model structure, e.g. autoregressive–moving-average (ARMA) models in the linear case, for which subspace methods turn out to be effective [59], [17]. Guarantees for the stability of the identified linear system are provided in [31] and [12]. For solving the deconvolution problem of finding the impulse response given the training data, regularization techniques are introduced in [40] and recent overviews are provided in [16] and [23]. As this problem also gained attention in the field of machine learning, kernel based techniques became prominent to identify the impulse response of linear systems, see [41] and [42].

For the nonlinear case, system identification techniques exist for Volterra series or Wiener-Hammerstein models [30], which consider a very limited structure of the model. Therefore, techniques from machine learning are alternatively being used, e.g. neural networks [35] or Gaussian process state-space models (GPSSMs), which this article focuses on. Using GPs for identification is first considered in [60] and [25] and is recently applied in predictive control schemes, e.g. [14]. A comprehensive overview on GP models for control is given in [24]. However, an analysis of the system stability is often missing in these studies.

Generally, the literature mainly utilizes two different interpretations of a GP [55]: The deterministic view, which takes the most likely value (the mean

function) as the output of the model and the stochastic view, which samples from a normal distribution as output of the model using the most likely value as the mean and the covariance function values as variance (which also serves as uncertainty measure of the model) [29]. A stability analysis for both interpretations is first performed in [5] and [4], respectively. [The work in \[1\] considers to enforce linear operator constraints into GP models.](#) Furthermore, [56] and [58] utilize GPs in a feedback linearizing scheme and [8] and [9] consider safe exploration of the state-space with an initially unknown dynamics. Nevertheless, it is not investigated how the assumption of system stability can be enforced during the learning of the model.

The dissipation of energy and the associated convergence of the dynamical system are a constraint for the model, which is highly relevant for many practical tasks. The area of robotics is a popular application of such constrained models, where autonomous dynamical system are employed to represent desired motions in programming by demonstrations tasks. After collecting training data of a goal-directed movement performed by a human, a stable nonlinear system is learned for a robust and time-independent reproduction, see e.g. [46]. As the convergence to the correct final position is important in goal-directed motions, enforcing stability to Gaussian mixture models (GMMs) is studied in [21] and [33] and more general techniques are developed in [22], [11], [43] and [36]. Further applications, where stability of the learned model is essential includes computer vision: Dynamics textures are learned from video data to generate sequences on a modified time scale [49]. However, videos, which are synthesized from a dynamic texture model are prone to degenerate due to instability of the underlying model [12].

While all these approaches consider the stability of the model, none of them faces the inherent challenge of data-driven approaches, that with finite data, the model is necessarily imprecise. The uncertainty which is introduced due to missing data is widely ignored. Therefore, this article develops a general framework, to model these uncertainties as stochastic component of a dynamical system and derives stability conditions for the model based on stochastic

stability tools from [26] and [28].

1.2. Contribution and structure

In this work, we introduce a novel identification scheme to learn stable GPSSMs. Using control Lyapunov functions, we augment the GPSSM to enforce asymptotic stability for the deterministic interpretation of the GP for an arbitrary dataset. For the stochastic interpretation, we derive conditions under which asymptotic convergence is almost surely achieved and show how additional training data can be used to ensure asymptotic stability if these conditions are not met on the initial dataset. For the control Lyapunov function, we propose a sum of squares (SOS) approach which learns the convergence behavior from data and which allows a computationally efficient estimation of multiple unknown equilibria. We illustrate the proposed method in comparison to other Lyapunov candidates.

Preliminary results have appeared in [57], which only shows the existence of an ultimate bound for the stochastic interpretation of the GP. In contrast, this article computes the size of the ultimate bound, shows under which conditions asymptotic stability is achieved, and how many additional training points are needed for asymptotic convergence. Furthermore, this article considers the estimation consistency of the model and multiple unknown equilibria, while [57] only deals with a single known equilibrium. We also provide more extensive simulation results.

The remainder of this paper is structured as follows: After formulating the considered problem in Sec. 2, Sec. 3 reviews GPSSMs and explains how equilibrium points are enforced in GP models. The main results for the stabilization of a GPSSM are shown in Sec. 4 including a derivation of the required number of training points. Section 5 investigates how control Lyapunov functions are learned from data to improve the identification, followed by numerical evaluation in Sec. 6 and a conclusion.

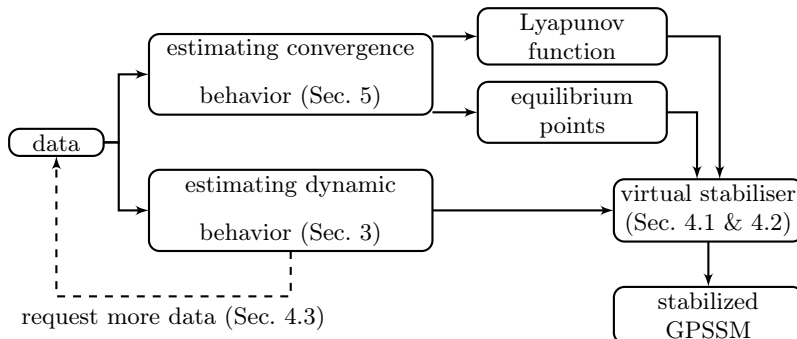


Figure 1: An overview of the proposed scheme for a data-driven stabilization of GPSSM.

1.3. Notation

Lower/upper case bold symbols denote vectors/matrices, $\mathbb{R}_{+,0}/\mathbb{R}_+$ all real positive numbers with/without zero, \mathbb{N}_0/\mathbb{N} all natural numbers with/without zero, $\lambda_{\min}(\cdot), \lambda_{\max}(\cdot)$ the minimal/maximal eigenvalue of a matrix and $\mathbb{E}[\cdot]/\mathbb{V}[\cdot]$ the expected value/variance of a random variable, respectively. \mathbf{I}_n denotes the $n \times n$ identity matrix, $\text{tr}(\cdot)$ the trace of a matrix, $\cdot \succ 0$ the positive definiteness of matrix or function and $\|\cdot\|$ the Euclidean norm if not state otherwise. $\sqrt{\cdot}$ applied to a matrix means is meant element-wise. The set of positive definite symmetric $n \times n$ matrices is denoted by \mathbb{S}_+^n . The colon in $1:N$ denotes all numbers $1, 2, \dots, N$. Accordingly, denotes $\mathbf{x}_{1:N}$ the concatenation $[\mathbf{x}_1^\top \ \dots \ \mathbf{x}_N^\top]^\top$ and $f(\mathbf{x}_{1:N})$ the concatenation of the function evaluations $[f(\mathbf{x}_1) \ \dots \ f(\mathbf{x}_N)]^\top$.

2. PROBLEM FORMULATION

Consider a discrete-time system with a continuous-valued state $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n$, with $n \in \mathbb{N}$, driven by the equation

$$\mathbf{x}_{\kappa+1} = \mathbf{f}(\mathbf{x}_\kappa), \quad (1)$$

with $\kappa \in \mathbb{N}$, $\mathbf{f} : \mathcal{X} \rightarrow \mathcal{X}$ and initial condition $\mathbf{x}_0 \in \mathcal{X}$. We assume $\mathbf{f}(\cdot)$ to be unknown, while the following assumptions are made:

Assumption 1. *There exist N_* unknown equilibria $\mathbf{x}_{i_*} \in \mathcal{X}_{i_*} \subseteq \mathcal{X}$, with $i_* = 1, \dots, N_*$ and $\mathbf{f}(\mathbf{x}_{i_*}) = \mathbf{x}_{i_*}$. Each of the equilibria is uniformly asymptotically stable with corresponding known domain of attraction $\mathcal{X}_{i_*} \subseteq \mathcal{X}$ for which holds*

$$\mathcal{X}_{i_*} \cap \mathcal{X}_{i'_*} = \emptyset, \quad \forall i'_* \neq i_*, \quad (2)$$

$$\mathcal{X}_1 \cup \mathcal{X}_2 \cup \dots \cup \mathcal{X}_{N_*} = \mathcal{X}. \quad (3)$$

This holds for many real world applications. For example in physical systems, it is often known that energy is dissipated and therefore the system eventually reaches an equilibrium point. The domains of attraction are also rather simple to classify e.g. consider a ball rolling to one or the other side of a hill. But finding the exact position of equilibria through observations is tedious as these are reached in general in infinite time. Therefore, we focus on the case that the equilibria \mathbf{x}_{i_*} are considered unknown and must be estimated from data. Furthermore, the following assumptions for $\mathbf{f}(\cdot)$ are made.

Assumption 2. *The function $\mathbf{f}(\cdot)$ is unknown, but assumed to be continuous.*

Continuity is a nonrestrictive assumption as it holds for a large class of systems. It is also necessary because generalization outside of the training dataset can not be expected for discontinuous dynamics [63].

Assumption 3. *Consecutive measurements of the state are available, thus N data pairs are given in the training set*

$$\mathcal{D} = \left\{ \left(\mathbf{x}^{(i)}, \mathbf{y}^{(i)} \right) \right\}_{i=1}^N, \quad (4)$$

where $\mathbf{y}^{(i)} = \mathbf{x}_{\kappa+1}^{(i)}$ is the consecutive state to $\mathbf{x}_{\kappa}^{(i)}$.

The data do not need to stem from a single trajectory. Each pair of consecutive states is not necessarily related to any other pair. An illustration is provided in Fig. 2

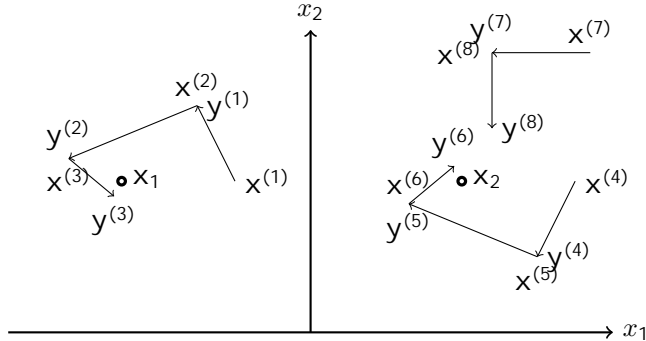


Figure 2: Illustration of a possible training dataset \mathcal{D} for the case of two equilibria $\mathbf{x}_{1*}, \mathbf{x}_{2*}$. Here the data stem from three different trajectories (1-2-3,4-5-6,7-8). Within one trajectory, the end point of one step is the starting point of the next, e.g. $\mathbf{y}^{(1)} = \mathbf{x}^{(2)}$, $\mathbf{y}^{(4)} = \mathbf{x}^{(5)}$, etc. But this is not the case across different trajectories.

The goal is to find a Gaussian process model $\hat{\mathbf{f}} : \mathcal{X} \rightarrow \mathcal{X}$ for the unknown true dynamics (1), denoted by

$$\mathbf{x}_{\kappa+1} = \mathbf{f}(\mathbf{x}_{\kappa}). \quad (5)$$

Modeling the unknown system (1) with a Gaussian process has several advantages: First, it allows to approximate any continuous function arbitrarily exactly [48], which is the case here according to Assumption 2. Second, a GP model provides, beside the most likely function value (the GP mean function), an uncertainty measure (the GP variance function), which encodes the precision of the model for any state $\mathbf{x} \in \mathcal{X}$. This model fidelity allows to apply risk-sensitive [34], uncertainty-aware [58] or path integral [53] control techniques.

Throughout the literature, different interpretations of GPs are considered (see Sec. 3.3 for a discussion). This article will focus on the two most relevant interpretations: The first here called the *deterministic GP* considers only the most likely estimate of $\mathbf{f}(\cdot)$, the GP mean function $\boldsymbol{\mu}(\cdot)$. For the second, the GP is interpreted as stochastic process (*probabilistic GP*) for which the uncertainty about the model, denoted by $\boldsymbol{\Sigma}(\cdot)$, leads to Gaussian distributed process noise. A detailed introduction to GPs are provided in Sec. 3.

For both interpretations, the prior knowledge in Assumption 1 regarding the

stability is generally not fulfilled by GP models [5], [4]. Therefore, this article proposes to augment the GP model by a virtual stabilizing command $\mathbf{u} : \mathcal{X} \rightarrow \mathbb{R}^n$. This $\mathbf{u}(\cdot)$ is not an external control input, but belongs internally to the model

$$\hat{\mathbf{f}}(\mathbf{x}_\kappa) := \mathbf{f}_{\mathcal{GP}}(\mathbf{x}_\kappa) + \mathbf{u}(\mathbf{x}_\kappa) \quad (6)$$

with the goal to ensure the same asymptotic behavior of the real system (1) and the model (5). In accordance with Assumption 1, this requires that the system model $\hat{\mathbf{f}}(\mathbf{x}_\kappa)$ has N_* asymptotically stable equilibrium points to match the asymptotic behavior of the real system. Since the equilibria \mathbf{x}_{i_*} of the true system in Assumption 1 are unknown, we consider estimated equilibria $\hat{\mathbf{x}}_{i_*}$ for which we show in Sec. 5 how they are determined from data.

For the *deterministic GP*, the goal is formulated as follows.

Objective 1. *Find a virtual control $\mathbf{u}(\cdot)$ such that the model*

$$\mathbf{x}_{\kappa+1} = \boldsymbol{\mu}(\mathbf{x}_\kappa) + \mathbf{u}(\mathbf{x}_\kappa) =: \bar{\mathbf{f}}(\mathbf{x}_\kappa), \quad (7)$$

has N_ equilibria $\hat{\mathbf{x}}_{i_*} \in \mathcal{X}_{i_*}$, which are asymptotically stable for the respective domain of attraction \mathcal{X}_{i_*} .*

So we aim for a dynamical system model, which has the same convergence properties as the true system (1). With the posterior mean function, we have chosen the most likely function approximation the GP model provides. However, since the training data is finite, we cannot expect a good model fit over the entire state space. We therefore also consider an extension of this *deterministic* model to a *probabilistic* GP model. Thus we introduce process noise into the model (which is not present in the true system) to ensure, that the model does not provide overconfident results in areas where training data is sparse. This leads to a stochastic dynamical system for which the following objective is formulated.

Objective 2. Find a virtual control $\mathbf{u}(\cdot)$ such that the probabilistic model

$$\mathbf{x}_{\kappa+1} = \bar{\mathbf{f}}(\mathbf{x}_\kappa) + \sqrt{\bar{\Sigma}(\mathbf{x}_\kappa)}\boldsymbol{\omega}_\kappa \quad \boldsymbol{\omega}_\kappa \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n), \quad (8)$$

has N_* equilibria $\hat{\mathbf{x}}_{i_*} \in \mathcal{X}_{i_*}$, which are almost surely asymptotically stable. Thus, $\forall \mathbf{x}_0 \in \mathcal{X}_{i_*}$ holds

$$\lim_{k \rightarrow \infty} \mathcal{P}(\|\mathbf{x}_k - \hat{\mathbf{x}}_{i_*}\| = 0) = 1, \quad \forall i_* = 1, \dots, N_* . \quad (9)$$

The probability space $(\Omega, \mathcal{F}, \mathcal{P})$ associated with the independent and identically distributed random variable $\boldsymbol{\omega}_\kappa$ has the sample space $\Omega = \mathbb{R}^n$, the sigma-algebra \mathcal{F} of Borel sets on Ω and the probability measure \mathcal{P} is a normal distribution.

This article shows how these objectives are achieved with an augmented GP model (6) by introducing an internal stabilizing command $\mathbf{u}(\cdot)$ into the GPSSM.

3. GAUSSIAN PROCESS STATE SPACE MODELS

3.1. Gaussian process regression

A Gaussian process is a stochastic process which assigns a jointly Gaussian distribution to any finite subset $\{\mathbf{x}_1, \dots, \mathbf{x}_M\} \subset \mathbb{R}^n$ in a continuous input domain [44]. It is often considered as a distribution over functions, denoted by

$$f_{\mathcal{GP}}(\mathbf{x}) \sim \mathcal{GP}(\mathbf{m}(\mathbf{x}), \mathbf{k}(\mathbf{x}, \mathbf{x}')) . \quad (10)$$

It is fully specified by its mean function $\mathbf{m}: \mathcal{X} \rightarrow \mathbb{R}$ and covariance function $\mathbf{k}: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$, which are parametrized through $\boldsymbol{\psi}$, considered as hyperparameters. If no structural prior knowledge for $\mathbf{f}(\cdot)$ is available, as it is the case for our problem setting, the mean function is commonly set to zero without loss of generality. A widely used covariance function is the squared exponential

(SE) kernel with automatic relevance determination

$$\mathfrak{k}_{\text{SE}}(\mathbf{x}, \mathbf{x}') = \zeta \exp \left(- \sum_{j=1}^n \frac{(x_j - x'_j)^2}{2l_j^2} \right), \quad (11)$$

where the hyperparameters $\boldsymbol{\psi} = [l_1^2 \ \dots \ l_n^2 \ \zeta]^\top$ are the signal variance $\zeta \in \mathbb{R}_{+,0}$ and the lengthscales $l_1^2, \dots, l_n^2 \in \mathbb{R}_+$. The SE kernel (11) is *universal* and therefore it allows to conclude that any continuous function $f(\mathbf{x})$ can be approximated arbitrarily exactly with the GP given sufficiently many training data points. For more details, we refer to the existing literature [2] and [45].

To model the dynamics (1) with a n -dimensional state space, n independent GPs are concatenated, denoted by

$$\mathbf{f}_{\mathcal{GP}}(\cdot) \sim \mathcal{GP}(\mathbf{m}(\cdot), \mathfrak{k}(\cdot, \cdot)) \equiv \begin{cases} f_{\mathcal{GP},1}(\cdot) \sim \mathcal{GP}(\mathbf{m}_1(\cdot), \mathfrak{k}_1(\cdot, \cdot)) \\ \vdots \\ f_{\mathcal{GP},n}(\cdot) \sim \mathcal{GP}(\mathbf{m}_n(\cdot), \mathfrak{k}_n(\cdot, \cdot)), \end{cases}$$

where $\mathbf{m}_j, \mathfrak{k}_j$ are the mean and kernel functions with corresponding hyperparameters $\boldsymbol{\psi}_j, j = 1, \dots, n$. The optimal ones are obtained through, a likelihood maximization based on Bayesian principles

$$\begin{aligned} \max_{\boldsymbol{\psi}_j} \log p \left(\mathbf{y}_j^{(1:N)} | \mathbf{x}^{(1:N)}, \boldsymbol{\psi}_j \right) = \\ \max_{\boldsymbol{\psi}_j} - \frac{1}{2} \left(\mathbf{y}_j^{(1:N)T} \mathbf{K}_j^{-1} \mathbf{y}_j^{(1:N)} + \log \det \mathbf{K}_j + N \log(2\pi) \right), \end{aligned} \quad (12)$$

where

$$\mathbf{K}_j = \begin{bmatrix} \mathfrak{k}_j(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \dots & \mathfrak{k}_j(\mathbf{x}^{(1)}, \mathbf{x}^{(N)}) \\ \vdots & \ddots & \vdots \\ \mathfrak{k}_j(\mathbf{x}^{(N)}, \mathbf{x}^{(1)}) & \dots & \mathfrak{k}_j(\mathbf{x}^{(N)}, \mathbf{x}^{(N)}) \end{bmatrix} \in \mathbb{R}^{N \times N},$$

which defines the notation $\mathfrak{k}_j(\mathbf{x}^{(1:N)}, \mathbf{x}^{(1:N)}) := \mathbf{K}_j$. The optimization (12) is

generally a non-convex problem, thus one cannot expect to obtain the optimal parameters in practice. However, each local maximum of the likelihood is considered as valid interpretation of the data [44]. For an analysis of the error in a GPSSM made by a suboptimal hyperparameter or kernel choice, we refer to [7].

To use the GPs for predictions at a test input $\mathbf{x} \in \mathcal{X}$, the dataset \mathcal{D} is used to infer the j -th component of the output $f_j(\mathbf{x})$, which is jointly Gaussian distributed with the training data

$$\begin{bmatrix} f_j(\mathbf{x}) \\ \mathbf{y}_j^{(1:N)} \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mathbf{m}_j(\mathbf{x}) \\ \mathbf{m}_j(\mathbf{x}^{(1:N)}) \end{bmatrix}, \begin{bmatrix} \mathfrak{k}_j(\mathbf{x}, \mathbf{x}) & \mathbf{k}_j^\top \\ \mathbf{k}_j & \mathbf{K}_j \end{bmatrix} \right), \quad (13)$$

where $\mathbf{k}_j = \mathfrak{k}_j(\mathbf{x}^{(1:N)}, \mathbf{x})$. Conditioning on the data \mathcal{D} yields

$$f_j(\mathbf{x}) | \mathcal{D}, \mathbf{x} \sim \mathcal{N} \left(\mathbf{m}_j(\mathbf{x}) + \mathbf{k}_j^\top \mathbf{K}_j^{-1} \left(\mathbf{y}_j^{(1:N)} - \mathbf{m}_j(\mathbf{x}^{(1:N)}) \right), \mathfrak{k}_j(\mathbf{x}, \mathbf{x}) - \mathbf{k}_j^\top \mathbf{K}_j^{-1} \mathbf{k}_j \right). \quad (14)$$

3.2. GP models with equilibria

We know from Assumption 1, that the true function $\mathbf{f}(\cdot)$ has exactly N_* equilibria at $\mathbf{x}_*^{(i)}$. Since the locations of these equilibria are unknown, we propose an estimation procedure to obtain these from the training data in Sec. 5. Given these estimates $\hat{\mathbf{x}}_{i_*}$, we must ensure the GP model has equilibria at these locations, thus $\hat{\mathbf{f}}(\hat{\mathbf{x}}_{i_*}) = \hat{\mathbf{x}}_{i_*}, \forall i_* = 1, \dots, N_*$. Therefore, these equilibria are added as training points $\mathcal{D}_* = \mathcal{D} \cup \bigcup_{i_*=1}^{N_*} (\hat{\mathbf{x}}_{i_*}, \hat{\mathbf{x}}_{i_*})$, which implicitly assumes that $\mathbf{m}(\hat{\mathbf{x}}_{i_*}) = \hat{\mathbf{x}}_{i_*}$. This enforces, that every function described by the GP has at least N_* equilibria at $\hat{\mathbf{x}}_{i_*}$, thus $\mathbf{f}_{\mathcal{GP}}(\hat{\mathbf{x}}_{i_*}) = \hat{\mathbf{x}}_{i_*}$. With

$$\begin{aligned} \tilde{\mathbf{y}}_j &= \begin{bmatrix} \mathbf{y}_j^{(1:N)\top} - \mathbf{m}_j(\mathbf{x}^{(1:N)}) & \mathbf{0}_{1 \times N_*} \end{bmatrix}^\top \in \mathbb{R}^{N+N_*} \\ \tilde{\mathbf{K}}_j &= \begin{bmatrix} \mathbf{K}_j & \mathfrak{k}_j(\mathbf{x}^{(1:N)}, \hat{\mathbf{x}}_{1:N_*}) \\ \mathfrak{k}_j(\mathbf{x}^{(1:N)}, \hat{\mathbf{x}}_{1:N_*}) & \mathfrak{k}_j(\hat{\mathbf{x}}_{1:N_*}, \hat{\mathbf{x}}_{1:N_*}) \end{bmatrix} \in \mathbb{R}^{(N+N_*) \times (N+N_*)}. \end{aligned}$$

and $\tilde{\mathbf{k}}_j = \begin{bmatrix} \mathbf{k}_j^\top & \mathbf{k}_j(\mathbf{x},) \end{bmatrix}^\top \in \mathbb{R}^{N+N_*}$, the updated mean and variance predictions for the outputs $\mathbf{f}(\mathbf{x})$ are defined as follows

$$f_j(\mathbf{x})|\mathcal{D}_*, \mathbf{x} \sim \mathcal{N}\left(\underbrace{\mathbf{m}_j(\mathbf{x}) + \tilde{\mathbf{k}}_j^\top \tilde{\mathbf{K}}_j^{-1} \tilde{\mathbf{y}}_j}_{:=\mu_j(\mathbf{x})}, \underbrace{\mathbf{k}_j(\mathbf{x}, \mathbf{x}) - \tilde{\mathbf{k}}_j^\top \tilde{\mathbf{K}}_j^{-1} \tilde{\mathbf{k}}_j}_{:=\sigma_j^2(\mathbf{x})}\right). \quad (15)$$

for $j = 1, \dots, n$. Concatenating them for the multiple dimensions leads to

$$\boldsymbol{\mu}(\mathbf{x}) = \begin{bmatrix} \mu_1(\mathbf{x}) \\ \vdots \\ \mu_n(\mathbf{x}) \end{bmatrix}, \boldsymbol{\Sigma}(\mathbf{x}) = \begin{bmatrix} \sigma_1^2(\mathbf{x}) & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \sigma_n^2(\mathbf{x}) \end{bmatrix}. \quad (16)$$

Due to the absence of any observation noise, the following conclusion regarding the mean and variance function of the GP are possible.

Lemma 1. *Consider a Gaussian process with SE kernel (11). Then, the concatenation (16) of the GP mean function (15) has equilibria at $\hat{\mathbf{x}}_{i_*}$, thus $\boldsymbol{\mu}(\hat{\mathbf{x}}_{i_*}) = \hat{\mathbf{x}}_{i_*}$, $\forall i_* = 1, \dots, N_*$. Furthermore, the posterior variance functions σ_j^2 , $j = 1, \dots, n$ defined in (15) is upper bounded by*

$$\sigma_j^2(\mathbf{x}) \leq \bar{\zeta} \left(1 - \exp\left(\frac{-\|\mathbf{x} - \hat{\mathbf{x}}_{i_*}\|^2}{\underline{l}^2}\right)\right) \quad \begin{array}{l} \forall j = 1, \dots, n \\ \forall \mathbf{x} \in \mathcal{X}, \forall i_*, \end{array}$$

where $\bar{\zeta} = \max_j \zeta_j$ is the largest of all signal variances of all GPs and $\underline{l}^2 = \min_{i,j} l_{i,j}^2$ is the smallest of all lengthscales $l_{i,j}^2$ across all GPs $i = 1, \dots, n$ and input dimensions $j = 1, \dots, n$. This implies $\sigma_j^2(\hat{\mathbf{x}}_{i_*}) = 0$, $\forall i_*$.

PROOF. The equilibria of the mean function follow from [58]. The variance function decreases point-wise globally with any additional training for the SE kernel (given constant hyperparameters), according to [56]. Therefore, the GP variance function for a dataset which contains the point $\hat{\mathbf{x}}_{i_*}$ can be upper bounded by the GP variance function for a dataset which *only* contains the point $\hat{\mathbf{x}}_{i_*}$ as

a single training point

$$\begin{aligned}\sigma_j^2(\mathbf{x}) &\leq \zeta_j \left(1 - \exp \left(- \sum_{i=1}^n \frac{(x_i - \hat{x}_{i_*,i})^2}{2l_{i,j}^2} \right)^2 \right) \\ &\leq \bar{\zeta} \left(1 - \exp \left(- \|\mathbf{x} - \hat{\mathbf{x}}_{i_*}\|^2 / \underline{l}^2 \right) \right) := \bar{\sigma}^2(\mathbf{x}),\end{aligned}$$

for all $j = 1, \dots, n$ which shows the proposed upper bound.

From Lemma 1, it is known, that the GP model (15), has at least the N_* equilibria at $\hat{\mathbf{x}}_{i_*}$. However, they might be unstable, their region of attraction might not match \mathcal{X}_{i_*} or additionally attractors might exist. Therefore, Objectives 1 and 2 are not fulfilled directly and Sec. 4 shows how they can be achieved. But before we quickly discuss the chosen interpretations of the GP model

3.3. Interpretations of GPs for dynamical systems

Besides the two presented interpretations of GPs for modeling dynamical system in (7) and (8), various other techniques exist. The overview in [55] presents - in addition to the two shown here - also a *robust interpretation* and the *belief space view*. The first is based on the analysis in [50] and is utilized in [8] and [6], but is limited to compact state-spaces and the required preconditions are difficult to verify. The latter is frequently employed in reinforcement learning approaches, e.g. [15] and [37], but is based on an approximation in the propagation to the next state and therefore not suitable for stability analysis. Furthermore, [18] introduces a GPSSM where a single function is sampled from the GP in a sequential fashion considering the covariance between the sample path and the model as discussed in [32]. This allows to consistently draw system with continuous deterministic transition functions. However, the resulting sequence of random variables is not a Markov process and therefore an analysis of its asymptotic behavior is more evolved. We therefore leave an analysis of this view on GP models for future work and focus on the two introduced - the *deterministic* and the *probabilistic* - interpretation.

4. STABILIZING GPSSMs

We will now show, how the deterministic and the probabilistic GPSSM in (7) and (8) are enforced to comply with the asymptotic stability requirement (Objectives 1 and 2) using a stabilizing command $\mathbf{u} \in \mathbb{R}^n$ which is internal, thus part of the model $\hat{\mathbf{f}}(\mathbf{x})$.

Consider N_* control Lyapunov functions $V_{\boldsymbol{\theta}_{i_*}}^{i_*} : \mathcal{X} \rightarrow \mathbb{R}_{+,0}$ which are each parameterized by $\boldsymbol{\theta}_{i_*}$ and the estimated equilibrium point $\hat{\mathbf{x}}_{i_*} \in \mathcal{X}_{i_*}$, for all of which hold the following properties are defined

Assumption 4. *The functions $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ are continuous and positive definite, thus*

$$V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}) > 0, \quad \forall \mathbf{x} \in \mathcal{X} \setminus \{\hat{\mathbf{x}}_{i_*}\} \quad \text{and} \quad V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\hat{\mathbf{x}}_{i_*}) = 0.$$

for all parameter choices $\boldsymbol{\theta}_{i_*} \in \Theta_{i_*}$, $\forall i_* = 1, \dots, N_*$.

Assumption 5. *The functions $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ are radially unbounded*

$$\lim_{\|\mathbf{x}\| \rightarrow \infty} V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}) = \infty, \quad \forall \boldsymbol{\theta}_{i_*} \in \Theta_{i_*}.$$

These are the standard conditions making all $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ Lyapunov candidates. In Sec. 5 we will explain how the parameters $\boldsymbol{\theta}_{i_*}$, $\hat{\mathbf{x}}_{i_*}$ are obtained from data, but first we discuss how the control Lyapunov functions $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ is used to compute the stabilizing command $\mathbf{u}(\cdot)$ for the GP model.

4.1. Deterministic case

For this case, the proposed next state is given by $\mathbf{x}_{\kappa+1} = \boldsymbol{\mu}(\mathbf{x}_\kappa)$. According to Lemma 1, there are equilibria at $\hat{\mathbf{x}}_{i_*}$, but they can generally be stable or unstable, and their domain of attractions is generally not \mathcal{X}_{i_*} . Therefore, a correction $\mathbf{u}(\cdot)$ is required in certain parts of the state-space as derived in Theorem 1.

Theorem 1. Consider the GP (16) with SE kernel (11) and the stabilizing command $\mathbf{u}^*(\cdot)$ obtained from the optimization

$$\mathbf{u}^*(\mathbf{x}_\kappa) = \arg \min_{\mathbf{u}} \frac{1}{2} \mathbf{u}^\top \mathbf{u}, \quad (17a)$$

$$\begin{aligned} \text{s.t. } V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\boldsymbol{\mu}(\mathbf{x}_\kappa) + \mathbf{u}) - V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}_\kappa) &< 0 && \text{if } \mathbf{x}_\kappa \neq \hat{\mathbf{x}}_{i_*}, \\ \text{and } \mathbf{u} = \mathbf{0} &&& \text{if } \mathbf{x}_\kappa = \hat{\mathbf{x}}_{i_*}, \end{aligned} \quad (17b)$$

where $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ is the i_* -th Lyapunov function with i_* being chosen such that $\mathbf{x}_0 \in \mathcal{X}_{i_*}$ and $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ fulfills Assumption 4. Then, the model

$$\mathbf{x}_{\kappa+1} = \bar{\mathbf{f}}(\mathbf{x}_\kappa) = \boldsymbol{\mu}(\mathbf{x}_\kappa) + \mathbf{u}^*(\mathbf{x}_\kappa), \quad (18)$$

converges uniformly asymptotically to the equilibrium $\hat{\mathbf{x}}_{i_*}$, for all initial values $\mathbf{x}_0 \in \mathcal{X}_{i_*}$.

PROOF. The function $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ is a valid Lyapunov candidate by Assumption 4 and the optimization is designed such that the Lyapunov function decreases in every step $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\bar{\mathbf{f}}(\mathbf{x}_\kappa)) - V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}_\kappa) < 0$, $\forall \mathbf{x}_\kappa \in \mathcal{X} \setminus \{\hat{\mathbf{x}}_{i_*}\}$. The optimization is feasible $\forall \mathbf{x}_\kappa \in \mathcal{X}$, since the solution $\mathbf{u} = \hat{\mathbf{x}}_{i_*} - \boldsymbol{\mu}(\hat{\mathbf{x}}_{i_*})$ is always in the constraint set because $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\hat{\mathbf{x}}_{i_*}) - V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}_\kappa) = -V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}_\kappa) < 0$.

This allows to conclude, that Objective 1 is achieved with the proposed augmented GP model $\bar{\mathbf{f}}(\cdot)$. A visualization of the stabilization through the optimization in (17) is shown in Fig. 3 on the left side.

Corollary 1. Let $N_* = 1$, the Lyapunov function is $V_{\boldsymbol{\theta}}$ radially unbounded (Assumption 5), and $\mathcal{X} = \mathbb{R}^n$. Furthermore, consider a GP (16) with SE kernel (11) and the stabilizing command $\mathbf{u}^*(\cdot)$ obtained from the optimization in (17). Then, the equilibrium $\hat{\mathbf{x}}_*$ of the model (18) is uniformly globally asymptotically stable.

PROOF. Radial unboundedness (Assumption 5) is according to [20] the only necessary additional criteria for global stability and therefore follows directly from Theorem 1.

We have chosen the optimization (17), because the the GPSSM represents the data optimal (according to the likelihood optimization) and should be distorted only minimal where necessary.

Furthermore, for an infinite number of training points on a compact state space, consistency of the proposed model (18) and the true system can be shown.

Proposition 1. *Consider the GP (16) on a compact set $\bar{\mathcal{X}} \subset \mathbb{R}^n$ with SE kernel (11) and Lyapunov functions $V_{\theta_{i_*}^{i_*}}(\cdot)$ which fulfill Assumption 4 and $V_{\theta_{i_*}^{i_*}}(\mathbf{f}(\mathbf{x}_\kappa)) - V_{\theta_{i_*}^{i_*}}(\mathbf{x}_\kappa) < 0, \forall \mathbf{x} \in \bar{\mathcal{X}} \setminus \{\hat{\mathbf{x}}_{i_*}\}, \forall i_*$. Assuming $\mathbf{f}(\cdot)$ being a sample from the GP from which infinitely many training points are generated using a dense distribution on $\bar{\mathcal{X}}$, then the model $\bar{\mathbf{f}}(\cdot)$ approaches the true function almost surely*

$$\mathcal{P} \left\{ \lim_{N \rightarrow \infty} \sup_{\mathbf{x} \in \bar{\mathcal{X}}} \|\mathbf{f}(\mathbf{x}_\kappa) - \bar{\mathbf{f}}(\mathbf{x}_\kappa)\| = 0 \right\} = 1 \quad (19)$$

for a stabilizing command $\mathbf{u}^*(\mathbf{x}_\kappa) = \mathbf{0} \forall \mathbf{x}_\kappa \in \bar{\mathcal{X}}$.

PROOF. Under the provided conditions, the difference between the mean function $\boldsymbol{\mu}(\cdot)$ and the true function $\mathbf{f}(\cdot)$ becomes arbitrarily small almost surely. This is a well established result from scattered data interpolation [47, Eq. 2.11], where the error is bounded by a power function (which corresponds to the posterior standard deviation of a GP [19, Sec. 5.2]), which tends to zero for $N \rightarrow \infty$ [27, Corollary 3.2.]. Since $V_{\theta_{i_*}^{i_*}}(\cdot)$ is a continuous function the condition

$$V_{\theta_{i_*}^{i_*}}(\boldsymbol{\mu}(\mathbf{x}_\kappa)) - V_{\theta_{i_*}^{i_*}}(\mathbf{x}_\kappa) < 0,$$

is fulfilled in the limit case $\forall \mathbf{x}_\kappa \in \bar{\mathcal{X}} \setminus \{\hat{\mathbf{x}}_{i_*}\}$, thus $\mathbf{u}^*(\mathbf{x}_\kappa) = \mathbf{0}, \forall \mathbf{x}_\kappa \in \bar{\mathcal{X}}$, which yields the provided result.

Thus, if the true stable system is perfectly learned by the GP, the stabilization is not required and does also not distort the model as desired.

Additionally, for convex Lyapunov functions, it can be shown that the stabilized GPSSM $\bar{\mathbf{f}}(\cdot)$ is never worse - in terms of prediction performance - than

the original, not stabilized GPSSM $\boldsymbol{\mu}(\cdot)$.

Proposition 2. *Consider the GP (16) with SE kernel (11) and convex Lyapunov functions $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ which fulfill Assumption 4 and $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{f}(\mathbf{x}_\kappa)) - V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}_\kappa) < 0, \forall \mathbf{x} \in \mathcal{X}_{i_*} \setminus \{\hat{\mathbf{x}}_{i_*}\}, \forall i_*$. Then, the prediction by the stabilized GPSSM $\bar{\mathbf{f}}(\cdot)$ is always closer (or equally close) to the true function $\mathbf{f}(\cdot)$ than the not stabilized GPSSM $\boldsymbol{\mu}(\cdot)$, thus*

$$\|\bar{\mathbf{f}}(\mathbf{x}_\kappa) - \mathbf{f}(\mathbf{x}_\kappa)\| \leq \|\boldsymbol{\mu}(\mathbf{x}_\kappa) - \mathbf{f}(\mathbf{x}_\kappa)\|, \quad \forall \mathbf{x}_\kappa \in \mathcal{X}. \quad (20)$$

PROOF. Since the true system (1) is asymptotically stable, the Lyapunov function $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ is decreasing with every step. Thus, for any $\mathbf{x}_\kappa \in \mathcal{X}$, the correct (but unknown) next state lies within the set $\mathcal{V} = \{\mathbf{x} \in \mathcal{X} \mid V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}) < V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}_\kappa)\}$, which is convex due to the convexity of $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$. For any $\mathbf{x}_\kappa \in \mathcal{V}$, $\mathbf{u}(\mathbf{x}_\kappa) = 0$ and therefore $\bar{\mathbf{f}}(\mathbf{x}_\kappa) = \boldsymbol{\mu}(\mathbf{x}_\kappa)$, which makes (20) hold with equality. For $\mathbf{x}_\kappa \notin \mathcal{V}$, the stabilized GPSSM corresponds to a projection on the convex set \mathcal{V} , thus

$$\bar{\mathbf{f}}(\mathbf{x}_\kappa) = \arg \min_{\mathbf{x}_{\kappa+1} \in \mathcal{V}} \|\mathbf{x}_{\kappa+1} - \boldsymbol{\mu}(\mathbf{x}_\kappa)\|. \quad (21)$$

This projection $\bar{\mathbf{f}}(\mathbf{x}_\kappa)$ is closer to any point in the convex set \mathcal{V} , than $\boldsymbol{\mu}(\mathbf{x}_\kappa)$, thus it is also closer to $\mathbf{f}(\mathbf{x}_\kappa)$, which concludes the proof.

The convexity of the Lyapunov functions also allows to conclude the following, which is important from a computational point of view.

Proposition 3. *The optimization problem (17) is convex if $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x})$ is convex.*

PROOF. In the constraint set (17b), $\boldsymbol{\mu}(\mathbf{x}_\kappa)$ and $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\mathbf{x}_\kappa)$ are constants with respect to \mathbf{u} whose addition/subtraction is convexity preserving, thus the set is convex if $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ is convex [13].

For non-convex $V_{\boldsymbol{\theta}_{i_*}}^{i_*}(\cdot)$ finding a global minimum cannot be expected. However, this is not critical concerning the stability, because any \mathbf{u} in the constraint set leads to a stable system. [Reaching](#) a local minimum will only result in behavior

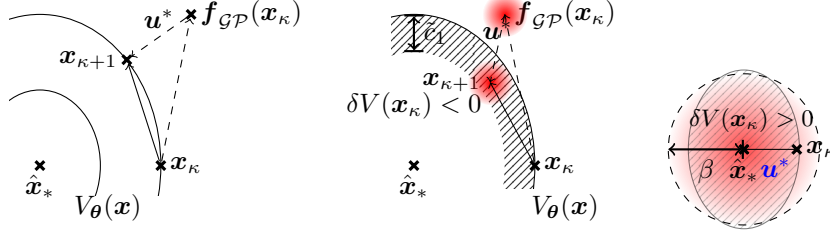


Figure 3: Illustration for the optimization (17) and (25) for the deterministic case and the probabilistic case on the left and in the middle, respectively. On the right, the infeasible situation for $n\bar{\zeta} \geq l^2$ in (25c) is visualized, thus $\mathbf{x}_\kappa \in \tilde{\mathcal{X}}_{i_*}$. The striped area indicates, the states \mathbf{x}_κ for which $\delta V(\mathbf{x}_\kappa) > 0$ for any \mathbf{u} .

different from the training data, but convergence is guaranteed. Nevertheless, for particular choices of the control Lyapunov function, the optimization (17) is convex, an overview is provided in Table 1. In case that the stability condition is already satisfied for the uncorrected GPSSM, the stabilizing command equals $\mathbf{0}$, thus the optimization will not be performed.

4.2. Probabilistic case

For the probabilistic case, the next state, according to the non-stabilized GP model, is given by $\mathbf{x}_{\kappa+1} = \mathbf{f}_{\mathcal{GP}}(\mathbf{x}_\kappa)$ and is a Gaussian distributed vector. To analyze the convergence behavior, we use tools of stochastic dynamical systems, reviewed in the following

Lemma 2 (Stability of stochastic systems [28]). *Consider a stochastic system of the form (8) and a positive definite function $V : \mathcal{X} \rightarrow \mathbb{R}_{+,0}$. If there exists a $V(\cdot)$ for which*

$$\mathbb{E}[V(\mathbf{x}_{\kappa+1})|\mathbf{x}_\kappa] - V(\mathbf{x}_\kappa) < 0, \quad \forall \mathbf{x}_\kappa \in \mathcal{X} \setminus \{\mathbf{x}_*\} \quad (22)$$

holds, then the equilibrium \mathbf{x}_ is almost surely uniformly asymptotically stable.*

If there exists a $V(\cdot)$ for which

$$\mathbb{E}[V(\mathbf{x}_{\kappa+1})|\mathbf{x}_\kappa] - V(\mathbf{x}_\kappa) < 0, \quad \forall \mathbf{x}_\kappa \in \mathcal{X} \setminus \mathcal{B}_\beta. \quad (23)$$

holds, then the system is almost surely uniformly ultimately bounded (UUB) to the hyperball

$$\mathcal{B}_\beta = \{\mathbf{x} \in \mathcal{X} \mid \|\mathbf{x} - \mathbf{x}_*\| < \beta\} \quad (24)$$

with ultimate bound $\beta > 0$.

This allows to conclude the following for the probabilistic interpretation of the GP model.

Theorem 2. Consider a GP (16) with SE kernel (11) and the stabilizing command $\mathbf{u}^*(\cdot)$ obtained from the optimization,

$$\mathbf{u}^*(\mathbf{x}_\kappa) = \arg \min_{\mathbf{u}} \frac{1}{2} \mathbf{u}^\top \mathbf{u}, \quad (25a)$$

$$\text{s.t. } \delta V(\mathbf{x}_\kappa) < 0 \quad \text{if } \mathbf{x}_\kappa \neq \hat{\mathbf{x}}_{i_*}, \quad (25b)$$

$$\delta V(\mathbf{x}_\kappa) = 0 \quad \text{if } \mathbf{x}_\kappa = \hat{\mathbf{x}}_{i_*},$$

$$\text{and } \mathbf{u}^* = \hat{\mathbf{x}}_{i_*} - \boldsymbol{\mu}(\mathbf{x}_\kappa) \quad \text{if } \delta V(\mathbf{x}_\kappa) > 0 \quad \forall \mathbf{u}, \quad (25c)$$

where $\delta V(\mathbf{x}_\kappa) = \mathbb{E} \left[V_{\boldsymbol{\theta}_{i_*}^{i_*}}^{i_*}(\mathbf{f}_{\mathcal{GP}}(\mathbf{x}_\kappa) + \mathbf{u}) \mid \mathbf{x}_\kappa \right] - V_{\boldsymbol{\theta}_{i_*}^{i_*}}^{i_*}(\mathbf{x}_\kappa)$, and $V_{\boldsymbol{\theta}_{i_*}^{i_*}}^{i_*}(\cdot)$ is the i_* -th Lyapunov function with i_* being chosen such that $\mathbf{x}_0 \in \mathcal{X}_{i_*}$ and fulfills Assumption 4. Then, the resulting model

$$\mathbf{x}_{\kappa+1} = \hat{\mathbf{f}}(\mathbf{x}_\kappa) := \mathbf{f}_{\mathcal{GP}}(\mathbf{x}_\kappa) + \mathbf{u}^*(\mathbf{x}_\kappa), \quad (26)$$

converges almost surely uniformly asymptotically to $\hat{\mathbf{x}}_{i_*}$, $\forall \mathbf{x}_0 \in \mathcal{X}_{i_*}$ if $n\bar{\zeta} < \underline{l}^2$. Otherwise, it converges almost surely uniformly to the hyperball

$$\mathcal{B}_\beta^{i_*} = \{\mathbf{x} \in \mathcal{X} \mid \|\mathbf{x} - \hat{\mathbf{x}}_{i_*}\| < \beta\} \quad (27)$$

with ultimate bound

$$\beta = \sqrt{\underline{l}^2 W_0 \left(\frac{n\bar{\zeta}}{-\underline{l}^2} e^{-n\bar{\zeta}/\underline{l}^2} \right) + n\bar{\zeta}}, \quad (28)$$

where $W_0 : [-e^{-1} \infty) \rightarrow [-1 \infty)$ is the principle branch of the Lambert W function.

PROOF. Similar to Theorem 1, $V_{\boldsymbol{\theta}_{i_*}^{i_*}}(\cdot)$ is a valid Lyapunov candidate by Assumption 4 and the optimization (25) is designed such that the expected value of the Lyapunov function decreases in every step $\delta V(\mathbf{x}_\kappa) < 0, \forall \mathbf{x}_\kappa \in \mathcal{X} \setminus \{\hat{\mathbf{x}}_{i_*}\}$. However, the optimization is infeasible for some \mathbf{x}_κ , because $V_{\boldsymbol{\theta}_{i_*}^{i_*}}(\mathbf{x}_\kappa) \rightarrow 0$ for $\mathbf{x}_\kappa \rightarrow \hat{\mathbf{x}}_{i_*}$ and $\mathbb{E}[V_{\boldsymbol{\theta}_{i_*}^{i_*}}(\mathbf{f}_{\mathcal{GP}}(\mathbf{x}_\kappa) + \mathbf{u}) | \mathbf{x}_\kappa] \neq 0$ in general for any \mathbf{u} . Thus, the set $\tilde{\mathcal{X}}_{i_*} = \{\mathbf{x}_\kappa \in \mathcal{X}_{i_*} | \delta V(\mathbf{x}_\kappa) > 0 \forall \mathbf{u}\}$ which denotes the states for which there exists no stabilizing command and therefore the constraint set $\delta V(\mathbf{x}_\kappa) < 0$ is empty. For all state outside of this set, $\mathbf{x} \in \mathcal{X} \setminus \tilde{\mathcal{X}}_{i_*}$, the conditions in Lemma 2 are fulfilled and the system converges to $\tilde{\mathcal{X}}_{i_*}$. The case $\mathbf{x}_\kappa \in \tilde{\mathcal{X}}_{i_*}$ is considered in (25c) and results in the stochastic dynamics

$$\mathbf{x}_{\kappa+1} \sim \mathcal{N}(\hat{\mathbf{x}}_{i_*}, \boldsymbol{\Sigma}(\mathbf{x}_\kappa)). \quad (29)$$

From now, we set $\mathbf{x}_{i_*} = \mathbf{0}$ for notational convenience without loss of generality and the quadratic Lyapunov function $V_{\text{sq}}(\mathbf{x}) = \mathbf{x}^\top \mathbf{x}$ is utilized for analysis. Note, that the control Lyapunov function $V_{\boldsymbol{\theta}_{i_*}^{i_*}}(\cdot)$ is only applied to ensure convergence to the set $\tilde{\mathcal{X}}_{i_*}$, but inside this set, it is no longer used, because - according to (25c) - \mathbf{u}^* is chosen independently of $V_{\boldsymbol{\theta}_{i_*}^{i_*}}(\cdot)$. Therefore, inside the set $\tilde{\mathcal{X}}_{i_*}$, the function $V_{\text{sq}}(\cdot)$ is utilized for analysis. The resulting $\delta V_{\text{sq}}(\cdot)$ is given by

$$\begin{aligned} \delta V_{\text{sq}}(\mathbf{x}_\kappa) &= \mathbb{E}[V_{\text{sq}}(\mathbf{x}_{\kappa+1}) | \mathbf{x}_\kappa] - V_{\text{sq}}(\mathbf{x}_\kappa) = \mathbb{E}[\mathbf{x}_{\kappa+1}^\top \mathbf{x}_{\kappa+1}] - \mathbf{x}_\kappa^\top \mathbf{x}_\kappa \\ &= \text{Tr}(\boldsymbol{\Sigma}(\mathbf{x}_\kappa)) - \mathbf{x}_\kappa^\top \mathbf{x}_\kappa = \sum_{j=1}^n \sigma_j^2(\mathbf{x}_\kappa) - \mathbf{x}_\kappa^\top \mathbf{x}_\kappa \\ &\leq \underbrace{n\bar{\zeta} \left(1 - \exp\left(\frac{-\|\mathbf{x}_\kappa\|^2}{l^2}\right)\right)}_{:= \underline{V}(\|\mathbf{x}_\kappa\|)} - \underbrace{\|\mathbf{x}_\kappa\|^2}_{:= \bar{V}(\|\mathbf{x}_\kappa\|)} \end{aligned}$$

using $\mathbb{E}[\mathbf{x}_{\kappa+1}] = \mathbf{0}$ from (29) and the upper bound introduced in Lemma 1.

Analyzing $\delta V_{\text{sq}}(\cdot)$, we obtain the obvious root at the origin $\delta V_{\text{sq}}(\mathbf{0}) = 0$ and

$$\lim_{\|\mathbf{x}_\kappa\| \rightarrow \infty} \delta V(\mathbf{x}_\kappa) = -\infty$$

due to the boundedness of \underline{V} . Thus, if there are no further roots outside the origin, we know the system is asymptotically stable ($\delta V(\mathbf{x}_\kappa)$ is negative definite) and if there are further roots, the outer most is the ultimate bound.

Comparing the derivatives of \bar{V}, \underline{V}

$$\frac{\partial \underline{V}(\|\mathbf{x}\|)}{\partial \|\mathbf{x}\|} = \frac{2n\bar{\zeta}}{\underline{l}^2} \exp\left(\frac{-\|\mathbf{x}\|^2}{\underline{l}^2}\right) \|\mathbf{x}\|, \quad \frac{\partial \bar{V}(\|\mathbf{x}\|)}{\partial \|\mathbf{x}\|} = 2\|\mathbf{x}\|,$$

it is clear, that \bar{V} increases $\forall \mathbf{x} \in \mathcal{X}$ faster than \underline{V} for $n\bar{\zeta}/\underline{l}^2 < 1$ (since $\exp(-\|\mathbf{x}\|^2/\underline{l}^2) \leq 1, \forall \mathbf{x} \in \mathcal{X}$). Thus, we conclude $\delta V(\mathbf{x}) < 0, \forall \mathbf{x} \in \mathcal{X} \setminus \{\mathbf{0}\}$ for $n\bar{\zeta} < \underline{l}^2$, which shows that the equilibrium is almost surely uniformly asymptotically stable.

For the case $n\bar{\zeta} > \underline{l}^2$, we use the fact, that difference $\frac{\partial \underline{V}(\|\mathbf{x}\|)}{\partial \|\mathbf{x}\|} - \frac{\partial \bar{V}(\|\mathbf{x}\|)}{\partial \|\mathbf{x}\|}$ changes its sign at most once in the positive domain. Thus, we only need to verify the proposed bound (28) as the single root of δV making use of the Lambert W function, more specifically its principle branch W_0 . Defining $z = \frac{n\bar{\zeta}}{-\underline{l}^2} e^{-n\bar{\zeta}/\underline{l}^2}$ and substituting $\|\mathbf{x}\| = \beta$ from (28) yields

$$\begin{aligned} \underline{V}(\beta) - \bar{V}(\beta) &= n\bar{\zeta} \left(1 - e^{-W_0(z) - n\bar{\zeta}/\underline{l}^2}\right) - \underline{l}^2 W_0(z) - n\bar{\zeta} \\ &= -n\bar{\zeta} e^{-W_0(z)} e^{-n\bar{\zeta}/\underline{l}^2} - \underline{l}^2 W_0(z) \\ &= \left(\frac{n\bar{\zeta}}{-\underline{l}^2} e^{-n\bar{\zeta}/\underline{l}^2} - W_0(z) e^{W_0(z)}\right) \underline{l}^2 e^{-W_0(z)} = (z - z) \underline{l}^2 e^{-W_0(z)} = 0, \end{aligned}$$

where the identity $W_0(z) e^{W_0(z)} = z$ known for the Lambert W function was used in the last step. Thus, we have shown that

$$\delta V(\mathbf{x}) < 0, \quad \forall \mathbf{x} \in \{\mathbf{x} \in \mathcal{X} \mid \|\mathbf{x}\| > \beta\} \quad (30)$$

which, according to Lemma 2, shows that (29) is almost surely uniformly ulti-

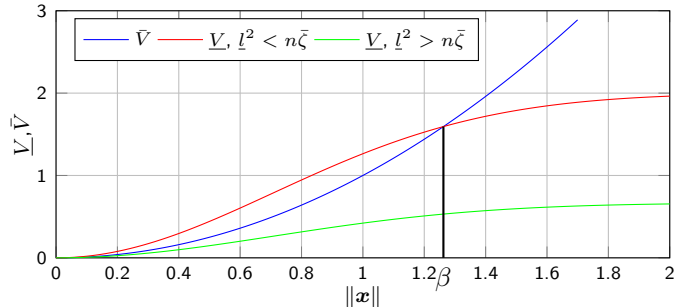


Figure 4: Comparing the functions $\bar{V}(\|\mathbf{x}\|)$ and $V(\|\mathbf{x}\|)$ for the two different cases of hyperparameters, resulting in almost sure UUB for $\underline{l}^2 < n\bar{\zeta}$ and almost sure uniform asymptotic stability for $\underline{l}^2 > n\bar{\zeta}$.

mately bounded with the specified bound.

Corollary 2. *Let $N_* = 1$, the Lyapunov function V_θ radially unbounded (Assumption 5), and $\mathcal{X} = \mathbb{R}^n$. Furthermore, consider a GP (16) with SE kernel (11) and the stabilizing command $\mathbf{u}^*(\cdot)$ obtained from the optimization in (25). Then, the equilibrium $\hat{\mathbf{x}}_*$ of the model (26) is almost surely globally uniformly asymptotically stable if $n\bar{\zeta} < \underline{l}^2$. For $n\bar{\zeta} \geq \underline{l}^2$ the system (26) is almost surely globally uniformly ultimately bounded to the set \mathcal{B}_β .*

PROOF. Radial unboundedness (Assumption 5) is according to [28] the only necessary additional criteria for the global convergence and therefore follows directly from Theorem 2.

A visualization of the stabilization through the optimization in (25) is shown in Fig. 3 in the center. The right side visualizes the non-feasible case in (29).

Remark 1. For any polynomial Lyapunov candidate, the moments of the random variable $V_\theta(\mathbf{f}_{\mathcal{GP}}(\mathbf{x}_\kappa) + \mathbf{u})$ are polynomial in the parameters of the normal distributed random variable $\mathbf{f}_{\mathcal{GP}}(\mathbf{x}_\kappa)$. For the computation of the expected value of such polynomials as required in (25b), see [54].

From Theorem 2 is concluded, that Objective 2 is only surely achieved if the hyperparameters in the likelihood optimization (12) are constrained to $n\bar{\zeta} < \underline{l}^2$.

However, limiting the search space of the hyperparameters results in a suboptimal model choice if the likelihood attains its maximum for $n\bar{\zeta} > \underline{l}^2$. For this case, Objective 2 is not achieved if the optimal hyperparameter choice is taken. Therefore, the following section investigates how additional measurements from the system must be taken to further to fulfill Objective 2.

4.3. Asymptotic convergence with additional data

Consider the hyperparameters are fixed and it holds $n\bar{\zeta} > \underline{l}^2$. Then, additional training data \mathcal{D}_{add} is necessary to achieve asymptotic stability. We therefore make the following additional assumption.

Assumption 6. *An additional dataset of the system (1), denoted by*

$$\mathcal{D}_{\text{add}} = \left\{ \left(\mathbf{x}^{(i)}, \mathbf{y}^{(i)} \right) \right\}_{i=N+1}^{N+N_{\text{add}}}, \quad (31)$$

is obtained for arbitrary input locations $\mathbf{x}^{(i)} \in \mathcal{B}_\beta$. The hyperparameters ψ_j , for all $j = 1, \dots, n$ of the GP model remain constant.

This section elaborates how many additional data points $N_{\text{add}} = |\mathcal{D}_{\text{add}}|$ are necessary to fulfill Objective 2 without making assumptions regarding the initial dataset \mathcal{D} .

As the following derivations are equivalent for all equilibria $\hat{\mathbf{x}}_{i^*}$, we consider only a single equilibrium at the origin $\hat{\mathbf{x}}_* = \mathbf{0}$, without loss of generality.

Theorem 3. *Consider a GP (16) with SE kernel (11) and the stabilizing command $\mathbf{u}^*(\cdot)$ from (25). Then, there exists a set of additional training data points \mathcal{D}_{add} of size*

$$N_{\text{add}} = (1 + 2\beta/\epsilon)^n + 1, \quad (32)$$

where $\epsilon := -\underline{l}^2 \log(1 - \tilde{r}^2/n\bar{\zeta})$ and

$$\tilde{r} := \frac{l}{3\sqrt{n\bar{\zeta}}} \sqrt{-n\bar{\zeta} W_{-1} \left(e^{\frac{\underline{l}^2}{n\bar{\zeta}} - 1} \left(\frac{\underline{l}^2}{n\bar{\zeta}} - 1 \right) \right) + \underline{l}^2 - n\bar{\zeta}}$$

such that for any (fixed) hyperparameter set $\boldsymbol{\psi}_j$, the system (26) is almost surely asymptotically stable.

PROOF. According to proof of Theorem 1, consider

$$\delta V_{\text{sq}}(\mathbf{x}_\kappa) = \sum_{j=1}^n \sigma_j^2(\mathbf{x}_\kappa) - \mathbf{x}_\kappa^\top \mathbf{x}_\kappa \leq n \max_j \sigma_j^2(\mathbf{x}_\kappa) - \mathbf{x}_\kappa^\top \mathbf{x}_\kappa,$$

which is negative for $\|\mathbf{x}_\kappa\| > \beta$. We first investigate, where a single additional data point must be placed to ensure $\delta V_{\text{sq}}(\mathbf{x}_\kappa) < 0$ near the origin.

We first introduce the notation $\mathfrak{k}(\mathbf{x}, \mathbf{x}') = \mathfrak{k}(\mathbf{x} - \mathbf{x}')$ (Since the SE kernel is stationary and can therefore be expressed as the difference of its arguments.) and $\sigma_{\mathcal{D}_2}^2(\cdot)$, which is used for the variance function with hyperparameters $\bar{\zeta}, \bar{l}^2$ and the dataset $\mathcal{D}_2 = \{(\mathbf{0}, \mathbf{0}), (\mathbf{x}^{(a)}, \mathbf{y}^{(a)})\}$. This is used as an upper bound for the variance function $\max_j \sigma_j^2(\cdot)$ with dataset $\mathcal{D}_2 \cup \mathcal{D}$, because the variance generally reduces with any additional data point [56].

First, we consider the one dimensional case $n = 1$ and compute the location for an additional data point $x^{(a)} = r$ such that $n\sigma_{\mathcal{D}_2}^2(x) < x^2$. The variance for two data points $x^{(0)} = 0, x^{(a)} = r$ is according to [61, Appendix A] given by

$$\sigma_{\mathcal{D}_2}^2(x) = \bar{\zeta} - \frac{\bar{\zeta} (\mathfrak{k}(x)^2 + \mathfrak{k}(r-x)^2) - 2\mathfrak{k}(r)\mathfrak{k}(x)\mathfrak{k}(r-x)}{\bar{\zeta}^2 - \mathfrak{k}(r)^2}, \quad (33)$$

where $\mathfrak{k}(0) = \bar{\zeta}$ is used. Multiplication of $\bar{\zeta}^2 - \mathfrak{k}(r)^2$ on both sides of the inequality $\sigma_{\mathcal{D}_2}^2(x) < x^2$ leads to the functions

$$M(r, x) := \bar{\zeta}^3 - \bar{\zeta}\mathfrak{k}(r)^2 - \bar{\zeta}\mathfrak{k}(x)^2 - \bar{\zeta}\mathfrak{k}(r-x)^2 - 2\mathfrak{k}(r)\mathfrak{k}(x)\mathfrak{k}(r-x), \quad (34)$$

$$N(r, x) := (\bar{\zeta}^2 - \mathfrak{k}(r)^2)x^2, \quad (35)$$

for the left and right side, respectively. Their derivatives are

$$\begin{aligned}\frac{\partial M(r, x)}{\partial x} &= \frac{2}{l^2} \left(r (\mathfrak{k}(r)\mathfrak{k}(x)\mathfrak{k}(r-x) - \bar{\zeta}\mathfrak{k}(r-x)^2) \right. \\ &\quad \left. + x (\bar{\zeta}\mathfrak{k}(x)^2 + \bar{\zeta}\mathfrak{k}(r-x)^2 - 2\mathfrak{k}(r)\mathfrak{k}(x)\mathfrak{k}(r-x)) \right), \\ \frac{\partial N(r, x)}{\partial x} &= 2x (\bar{\zeta}^2 - \mathfrak{k}(r)^2).\end{aligned}$$

As both vanish at the origin, we further consider the second derivative evaluated at $x = 0$

$$\begin{aligned}\tilde{M}(r) &:= \frac{\partial^2 M(r, 0)}{\partial x^2} = \frac{2}{l^2} \left(\bar{\zeta}^3 - \bar{\zeta}\mathfrak{k}(r)^2 \left(1 + \frac{r^2}{l^2} \right) \right), \\ \tilde{N}(r) &:= \frac{\partial^2 N(r, 0)}{\partial x^2} = 2(\bar{\zeta}^2 - \mathfrak{k}(r)^2).\end{aligned}$$

Using the lower branch of the Lambert- W function, denoted by W_{-1} , one can show according to [62] that

$$n\tilde{M}(r) - \tilde{N}(r) = \frac{2}{l^2} \left(n\bar{\zeta}^3 - \bar{\zeta}^2 l^2 - \mathfrak{k}(r)^2 \left(n\bar{\zeta} \left(1 + \frac{r^2}{l^2} \right) - l^2 \right) \right)$$

has only one positive zero at

$$r = \frac{l}{\sqrt{n\bar{\zeta}}} \sqrt{-n\bar{\zeta}W_{-1} \left(e^{\frac{l^2}{n\bar{\zeta}} - 1} \left(\frac{l^2}{n\bar{\zeta}} - 1 \right) \right) + l^2 - n\bar{\zeta}}.$$

This allows to conclude that $nM(r, x) - N(r, x) < 0$ and thereby $\delta V_{\text{sq}}(x) < 0$ holds for all $x \in (0, x^{(a)})$ if $x^{(a)} < r$.

Generalizing this result to multiple dimensions $\|\mathbf{x}^{(a)}\| < r$ allows only to conclude, that $\delta V_{\text{sq}}(\mathbf{x}) < 0$ for all $\mathbf{x} = \alpha \mathbf{x}^{(a)}$ with $0 < \alpha \leq 1$. Thus, it only holds on a line in the \mathbb{R}^n , which is visualized in Fig. 6 on the left.

We therefore analyze the variance $\sigma_{D_2}^2(\mathbf{x})$ as \mathbf{x} is moved on a sphere centered at the origin. Defining the constants $c = \mathfrak{k}(\|\mathbf{x}\|)$, $d = \mathfrak{k}(\|\mathbf{x}^{(\bar{a})}\|)$ and the

variable $\tau = \|\mathbf{x} - \mathbf{x}^{(\bar{a})}\|$, the derivative of $\sigma_{\mathcal{D}_2}^2(\tau)$ with respect to τ is given by

$$\frac{\partial \sigma_{\mathcal{D}_2}^2(\tau)}{\partial \tau} = \frac{\partial \mathfrak{k}(\tau)}{\partial \tau} \frac{-\bar{\zeta}}{\bar{\zeta}^2 - d^2} (2\mathfrak{k}(\tau) - 2cd).$$

It shows, that the gradient is non-negative, because $\frac{\partial \mathfrak{k}(\tau)}{\partial \tau} \leq 0$ for the SE kernel and $\mathfrak{k}(\tau) \geq cd$

$$\begin{aligned} \exp\left(\frac{-\tau^2}{2l^2}\right) &\geq \exp\left(\frac{-\|\mathbf{x}\|^2}{2l^2}\right) \exp\left(\frac{-\|\mathbf{x}^{(\bar{a})}\|^2}{2l^2}\right) \\ \|\mathbf{x} - \mathbf{x}^{(\bar{a})}\|^2 &\leq \|\mathbf{x}\|^2 + \|\mathbf{x}^{(\bar{a})}\|^2 \end{aligned}$$

according to the triangle inequality. Thus, the point with the highest variance on the sphere lies on the opposite side of the training point

$$\sigma_{\mathcal{D}_2}^2(-\mathbf{x}^{(\bar{a})}) = \max_{\|\mathbf{x}\|=\|\mathbf{x}^{(\bar{a})}\|} \sigma_{\mathcal{D}_2}^2(\mathbf{x}),$$

which is illustrated in Fig. 6 in the center. Thus, $\mathbf{x}^{(\bar{a})}$ must be close enough to the origin, such that there exists a \tilde{r} for which all points $\|\mathbf{x}\| < \tilde{r}$ are closer to $\mathbf{x}^{(\bar{a})}$, then to $\mathbf{x}^{(a)}$,

$$\begin{aligned} \|\mathbf{x}^{(\bar{a})} + \mathbf{x}\| &\leq \|\mathbf{x}^{(\bar{a})}\| + \|\mathbf{x}\| \leq \|\mathbf{x}^{(a)} - \mathbf{x}\| = \|\mathbf{x}^{(a)}\| - \|\mathbf{x}\| \\ \|\mathbf{x}^{(\bar{a})}\| + 2\|\mathbf{x}\| &\leq \|\mathbf{x}^{(\bar{a})}\| \quad \Rightarrow \quad \|\mathbf{x}^{(\bar{a})}\| \leq \frac{1}{3}\|\mathbf{x}^{(a)}\| \end{aligned}$$

accordingly, $\tilde{r} := r/3$. Extending the result from the one dimensional case, it is shown that $n\sigma_{\mathcal{D}_2}^2(\mathbf{x}) < \|\mathbf{x}\|^2$ for all \mathbf{x} with $0 < \|\mathbf{x}\| < \tilde{r}$ if $\|\mathbf{x}^{(\bar{a})}\| \leq \tilde{r}$. Thus we set $\mathbf{x}^{(N+1)} = \mathbf{x}^{(\bar{a})}$ according to Assumption 6.

In general, $\tilde{r} < \beta$, thus for \mathbf{x} with $\tilde{r} \leq \|\mathbf{x}\| \leq \beta$, it was not shown so far, that $n\sigma_{\mathcal{D}_2}^2(\mathbf{x}) < \|\mathbf{x}\|^2$ holds. Nevertheless, it is known, that $\|\mathbf{x}\|^2 \leq \tilde{r}^2$ holds in this region. Thus, we next show how $n\sigma_{\mathcal{D}_2}^2(\mathbf{x}) < \tilde{r}^2$ is ensured for $\|\mathbf{x}\| < \beta$. Using the derived bound in Lemma 1, we can show, that every additional data point $\mathbf{x}^{(i)}$, $i \in [N+2 \ N + N_{\text{add}}]$ upper bounds the variance function in a radius ϵ

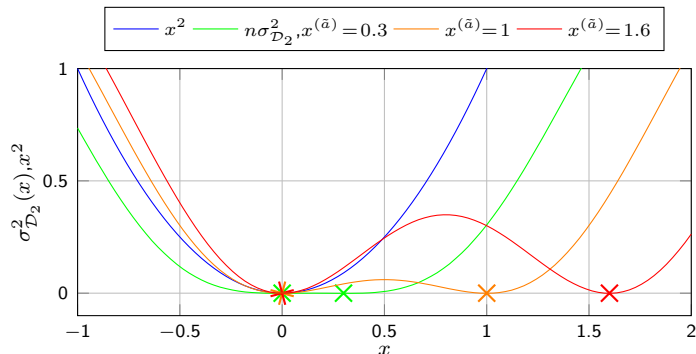


Figure 5: Numerical illustration for r as derived in the proof of Theorem 3 in the one dimensional case. For $\bar{\zeta} = 2$, $\underline{l}^2 = 1$, it can be seen that $x^2 > \sigma_{\mathcal{D}_2}^2(x)$ holds $\forall x$ only for $x^{(\bar{a})} = 0.3 < \bar{r}$ (green). For $x^{(\bar{a})} = 1$ (orange) this is only ensured for $0 < x < 1$, for $x^{(\bar{a})} = 1.6$, this is nowhere guaranteed.

as given by

$$\bar{r}^2 > n\bar{\zeta} \left(1 - \exp \left(\frac{-\|\mathbf{x} - \mathbf{x}^{(i)}\|^2}{\underline{l}^2} \right) \right) \Rightarrow \|\mathbf{x} - \mathbf{x}^{(i)}\|^2 < \epsilon. \quad (36)$$

The covering number, which indicates how many spheres with radius ϵ are required to cover a sphere with radius β is upper bounded according to [52] by $(1 + 2\beta/\epsilon)^n$. Thus at most $(1 + 2\beta/\epsilon)^n + 1$ data points are required to ensure $\delta V_{\text{sq}}(\mathbf{x}_\kappa) < 0 \forall \mathbf{x} \setminus \{\mathbf{0}\}$ which concludes the proof for almost surely asymptotic stability.

Example 1. For illustration, we consider three different cases of the system (29) simulated in two dimensions ($n = 2$) with $\underline{l}^2 = 1$: i) We set $\bar{\zeta} = 1/3$, which yields $n\bar{\zeta} \leq \underline{l}^2$ and thereby - according to Theorem 1 - leads to almost sure uniform asymptotic stability. ii) We choose $\bar{\zeta} = 1$, which results in $n\bar{\zeta} > \underline{l}^2$ and therefore almost sure UUB. iii) For the choice, $\bar{\zeta} = 1$, we add additional data points according to Assumption 6 and thereby establish asymptotic convergence.

The simulations start from the same 10 randomly sampled (uniformly from the interval $[-3 \ 3]^2$) initial points and draw in each step from an unbiased normal distribution with state dependent variance of a GP (15) with the only

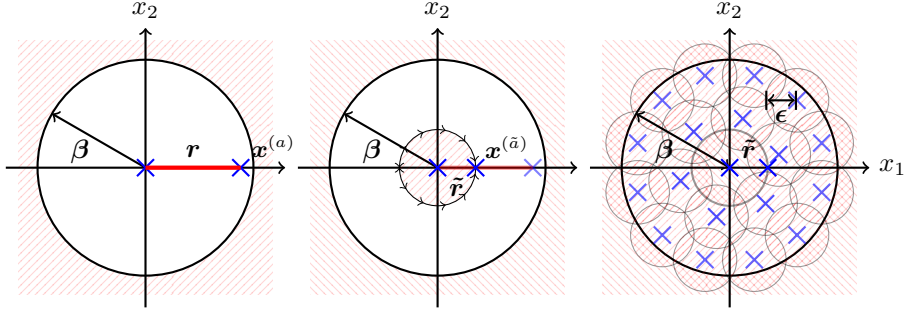


Figure 6: The illustrations show the steps of the proof for Theorem 3. Blue crosses indicate training data in \mathcal{D}_{add} and red areas the sets of \mathbf{x} for which $\delta V_{\text{sq}}(\mathbf{x}_\kappa) < 0$ is shown. The left drawing illustrates the one dimensional case. The middle drawing shows the extension from one dimension to multiple dimension, by choosing a $\|\mathbf{x}^{(\tilde{a})}\| = \tilde{r} < r/3$. The point $-\mathbf{x}^{(\tilde{a})}$ is the most critical point, as the variance $\sigma_{\mathcal{D}_2}^2$ is constantly decreasing along the circle as shown by the arrows. The right drawing illustrates a possible covering using an additional training points.

training point $(\mathbf{0}, \mathbf{0})$, thus

$$\mathbf{x}_{\kappa+1} \sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_2^2(\mathbf{x}_\kappa) & 0 \\ 0 & \sigma_1^2(\mathbf{x}_\kappa) \end{bmatrix} \right),$$

where $\sigma_j^2(\cdot)$ is the GP posterior variance function.

Figure 7 shows that trajectories for case i) quickly converge, while the states in case ii) drop close to the ultimate bound β quickly, but do not decrease further. The ultimate bound is exceeded irregularly, which is valid since the bound holds with probability 1 only for $\kappa \rightarrow \infty$. In case iii), where additional data points have been added to reduce the uncertainty $\sigma^2(\mathbf{x}_\kappa)$ the trajectories converge to zero again.

The hyperparameters lead to $\tilde{r} \approx 0.37$, $r \approx 1.12$ and $\epsilon \approx 0.0724$ from Theorem 3. This results in 1290 additional data points which we distributed randomly on the disk $\|\mathbf{x}\| < \beta$, since finding the particular covering is not a trivial problems. But, because the derivation includes conservatism, we can verify, that the random covering is sufficient to ensure $n\sigma^2(\mathbf{x}) < \|\mathbf{x}\|^2$ for all \mathbf{x} also for this randomized covering.

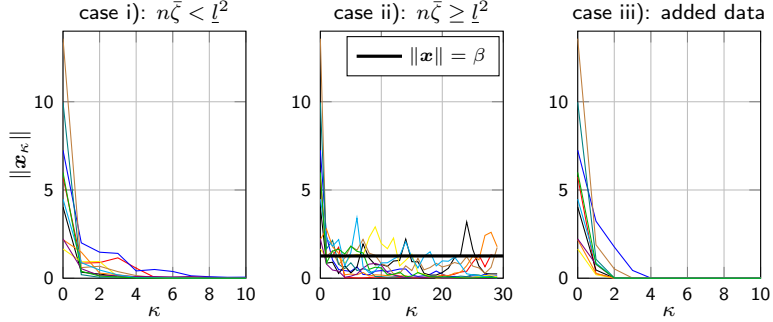


Figure 7: Illustration of Example 1 showing the norm of 10 trajectories for case i) asymptotic stability without any additional data points $n\bar{\zeta} < l^2$ (left) case ii) ultimate boundedness $n\bar{\zeta} \geq l^2$ (middle) and case iii) asymptotic stability for $n\bar{\zeta} \geq l^2$ with 1290 additional data points (right) according to Theorem 3.

5. LEARNING LYAPUNOV FUNCTIONS FROM DATA

To stabilize the GPSSM as just explained, an arbitrary control Lyapunov function, which fulfills Assumptions 4 and 5 can be utilized to compute the stabilization command $\mathbf{u}(\cdot)$. However, we want to avoid distorting the GPSSM whenever possible, because the GPSSM $\mathbf{f}_{\mathcal{GP}}(\cdot)$ is fitted to the data and is therefore the best available estimate to the true system. The following section derives a technique to adapt the control Lyapunov function $V_{\theta^{i_*}}^{i_*}(\cdot)$ properly - based on the available data.

5.1. General formulation

For fitting the Lyapunov functions $V_{\theta^{i_*}}^{i_*}(\cdot)$ to the training set \mathcal{D} by varying its parameters $\theta_{i_*} \in \Theta_{i_*}$ and $\hat{\mathbf{x}}_{i_*} \in \mathcal{X}_{i_*}$ the following is concluded.

Proposition 4. *Consider the N_* Lyapunov candidates $V_{\theta^{i_*}}^{i_*}(\cdot)$ under Assumptions 4 and 5, the dataset in Assumption 3 and the model $\bar{\mathbf{f}}(\cdot)$ from Theorem 1. If the optimization*

$$\eta_{i_*} := \min_{\substack{\theta_{i_*} \in \Theta_{i_*} \\ \hat{\mathbf{x}}_{i_*} \in \mathcal{X}_{i_*}}} \sum_{i \in \mathcal{I}_{i_*}} g \left(V_{\theta_{i_*}}^{i_*}(\mathbf{y}^{(i)}) - V_{\theta_{i_*}}^{i_*}(\mathbf{x}^{(i)}) \right), \quad (37)$$

where $\mathcal{I}_{i_*} = \{i = 1, \dots, N \mid \mathbf{x}^{(i)} \in \mathcal{X}_{i_*}\}$ and where $g: \mathbb{R} \rightarrow \mathbb{R}$ is any function of the form

$$g(\xi) = \begin{cases} 0 & \text{for } \xi \leq 0 \\ \tilde{g}(\xi) & \text{for } \xi > 0 \end{cases} \quad \text{with } \tilde{g}(\xi) > 0 \quad \forall \xi. \quad (38)$$

results in $\eta_{i_*} = 0$, then the GP model $\bar{\mathbf{f}}(\mathbf{x})$, obtained from the stabilization in Theorem 1, is exact at the training data in the i_* -th region of attraction \mathcal{X}_{i_*} , thus

$$\left\| \bar{\mathbf{f}}(\mathbf{x}^{(i)}) - \mathbf{y}^{(i)} \right\| = 0, \quad \forall i \in \mathcal{I}_{i_*}. \quad (39)$$

PROOF. With $g(\cdot)$ defined in (38), it is clear that $\eta_{i_*} = 0$ holds if and only if

$$V_{\boldsymbol{\theta}_{i_*}^{i_*}}(\mathbf{y}^{(i)}) - V_{\boldsymbol{\theta}_{i_*}^{i_*}}(\mathbf{x}^{(i)}) \leq 0, \quad \forall i \in \mathcal{I}_{i_*}. \quad (40)$$

According to Lemma 1, $\boldsymbol{\mu}(\mathbf{x}^{(i)}) = \mathbf{y}^{(i)}$. Therefore, at the training points $\mathbf{x}^{(i)}$, the constraint (17b) is fulfilled for $\mathbf{u} = \mathbf{0}$, which results in

$$\bar{\mathbf{f}}(\mathbf{x}^{(i)}) = \boldsymbol{\mu}(\mathbf{x}^{(i)}) = \mathbf{y}^{(i)}, \quad \forall i = 1, \dots, N$$

and concludes the proof.

Since the data points are recorded from a stable system (Assumption 1), there always exists a positive definite function $V: \mathcal{X}_{i_*} \rightarrow \mathbb{R}_{+,0}$ for which η_{i_*} as defined in (37) is zero. However, the parameterization $V_{\boldsymbol{\theta}_{i_*}^{i_*}}(\cdot)$ might not be chosen properly and therefore (40) might not hold for all data points. Nevertheless, the convergence properties as shown in Theorems 1 and 2 hold irrespectively, since they are based on Assumptions 4 and 5, which holds $\forall \boldsymbol{\theta} \in \Theta$. The suboptimality rather affects the accuracy of the identification, due to a non-zero distortion of the GP model as shown in Proposition 4.

The general definition of $g(\xi)$ ensures that any violation of condition (40),

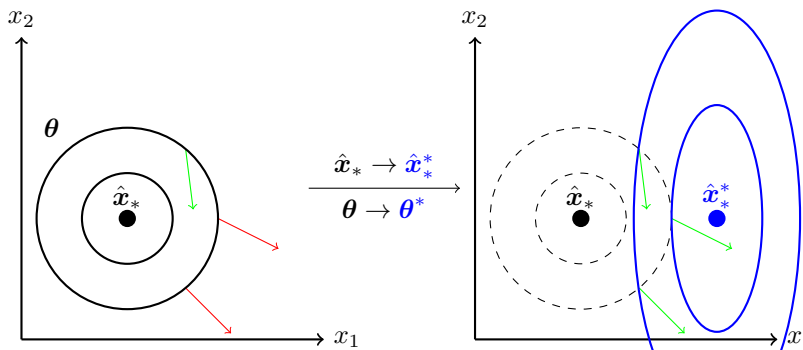


Figure 8: Illustration of the optimization (37) to search for a suitable Lyapunov function where $\hat{\mathbf{x}}_*$ and $\boldsymbol{\theta}^*$ denote the optimal values for $\hat{\mathbf{x}}_*$ and $\boldsymbol{\theta}$, respectively. The solid/dashed black/blue lines are boundaries of the level sets of the Lyapunov candidate, red arrows indicate violation of condition (40), green arrows accordance to condition (40).

which results in $\xi > 0$ is penalized, while any data point which is consistent with (40) will not contribute to the objective function. The particular form of the function $\tilde{g}(\cdot)$ can be chosen by the designer, however some choices allow to solve (37) efficiently, as shown later. A visualization of this optimization is shown in Fig. 8.

5.2. Specific Lyapunov candidates

To be more specific, we review some possible Lyapunov candidates. For notational convenience, we write $V_{\boldsymbol{\theta}}$ for $V_{\boldsymbol{\theta}^{i_*}}^{i_*}(\cdot)$, $\hat{\mathbf{x}}_*$ for $\hat{\mathbf{x}}_{i_*}$, $\boldsymbol{\theta}$ for $\boldsymbol{\theta}_{i_*}$ and Θ for Θ_{i_*} .

Quadratic Lyapunov function. The first Lyapunov candidate is defined by

$$V_{\text{Sq}}(\mathbf{x}) = (\mathbf{x} - \hat{\mathbf{x}}_*)^\top \mathbf{P} (\mathbf{x} - \hat{\mathbf{x}}_*), \quad \mathbf{P} \succ 0, \quad (41)$$

thus $\boldsymbol{\theta} = \mathbf{P}$ and $\Theta = \mathbb{S}_+^n$. The following holds.

Proposition 5. *Given a quadratic Lyapunov candidate (41), a convex set \mathcal{X} and a non-decreasing convex function $g(\cdot)$, the optimization (37) is a biconvex optimization problem, thus convex in $\boldsymbol{\theta}$ for constant $\hat{\mathbf{x}}_*$ and convex in $\hat{\mathbf{x}}_*$ for constant $\boldsymbol{\theta}$.*

PROOF. For constant $\hat{\mathbf{x}}_*$, V_{Sq} is linear in $\boldsymbol{\theta}$. As the difference of two linear function is again linear, the argument of $g(\cdot)$ is linear in $\boldsymbol{\theta}$. For constant $\boldsymbol{\theta}$, V_{Sq} is quadratic in $\hat{\mathbf{x}}_*$, but the difference cancels the squared terms and thus the argument of $g(\cdot)$ is linear in $\hat{\mathbf{x}}_*$. The composition of a linear function and the non-decreasing convex function $g(\cdot)$ is again convex [13]. Since the sum of convex functions is convex and so are the constraint sets, the optimization problem is biconvex.

Therefore, the parameters of V_{Sq} can be determined efficiently. However, due to the limited flexibility of quadratic Lyapunov functions with very few parameters, η in (37) is generally larger for the quadratic Lyapunov function than for Lyapunov functions with more parameters, which are considered in the following.

Sum of squares Lyapunov functions. Sum of squares (SOS) [38] considers more general polynomial Lyapunov candidates defined as follows:

Definition 1. For $\mathbf{x} \in \mathbb{R}^n$, a multivariate polynomial $p(\mathbf{x})$ is a *sum of squares* if there exist some polynomials $r_m(\mathbf{x})$, $m = 1 \dots \mathcal{M}$ such that

$$p(\mathbf{x}) = \sum_{m=1}^{\mathcal{M}} r_m^2(\mathbf{x}). \quad (42)$$

An equivalent characterization of SOS is given as follows [39].

Property 1. A polynomial $p(\mathbf{x})$ of degree $2\mathcal{M}$ is a SOS if and only if there exists a positive semidefinite matrix $\mathbf{Q} \succeq 0$ and a vector of monomials² $\mathbf{m}(\mathbf{x}) \in \mathbb{R}^{\mathcal{M}_n}$ containing monomials of degree greater zero and less or equal \mathcal{M} , such that

$$p(\mathbf{x}) = \mathbf{m}(\mathbf{x})^\top \mathbf{Q} \mathbf{m}(\mathbf{x}). \quad (43)$$

²A monomial of degree \mathcal{M} in $\mathbf{x} \in \mathbb{R}^n$ is a scalar function $\prod_{j=1}^n x_j^{\alpha_j}$ where α_j are nonnegative integers with $\sum_{j=1}^n \alpha_j = \mathcal{M}$. The number of all possible monomials of degree $\leq \mathcal{M}$ and > 0 is $\mathcal{M}_n = \frac{(n+\mathcal{M})!}{n!\mathcal{M}!} - 1$.

This property reduces the construction of a SOS to finding the elements of the $\mathcal{M}_n \times \mathcal{M}_n$ dimensional matrix \mathbf{Q} . In order to make the SOS a valid Lyapunov candidate, we restrict the matrix to be positive definite and symmetric $\mathbf{Q} \in \mathbb{S}_+^{\mathcal{M}_n}$, thus

$$V_{\text{SOS}}(\mathbf{x}) = \mathbf{m}(\mathbf{x} - \hat{\mathbf{x}}_*)^\top \mathbf{Q} \mathbf{m}(\mathbf{x} - \hat{\mathbf{x}}_*), \quad \mathbf{Q} \succ 0, \quad (44)$$

thus $\boldsymbol{\theta} = \mathbf{Q}$ and $\Theta = \mathbb{S}_+^{\mathcal{M}_n}$. The following holds.

Proposition 6. *Given a sum of squares Lyapunov candidate (44) and a non-decreasing convex function $g(\cdot)$, the optimization (37) is convex in $\boldsymbol{\theta}$ for a fixed $\hat{\mathbf{x}}_*$.*

PROOF. The proof is analog to Proposition 5.

Thus by gaining more flexibility in the Lyapunov candidate, we have lost the computational efficiency to estimate the equilibrium point $\hat{\mathbf{x}}_*$.

Weighted sum of asymmetric quadratic functions (WSAQF). The WSAQF Lyapunov function, proposed in [22] as

$$V_{\text{WSAQF}}(\mathbf{x}) = (\mathbf{x} - \hat{\mathbf{x}}_*)^\top \mathbf{P}_0 (\mathbf{x} - \hat{\mathbf{x}}_*) + \sum_{l=1}^L \beta_l(\mathbf{x} - \hat{\mathbf{x}}_*) ((\mathbf{x} - \hat{\mathbf{x}}_*)^\top \mathbf{P}_l (\mathbf{x} - \hat{\mathbf{x}}_* - \boldsymbol{\xi}_l))^2, \\ \text{with } \beta_l(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x}^\top \mathbf{P}_l (\mathbf{x} - \boldsymbol{\xi}_l) \geq 0 \\ 0 & \text{otherwise,} \end{cases} \quad (45)$$

has continuous first order partial derivatives and is positive definite for $\mathbf{P}_0, \dots, \mathbf{P}_L \in \mathbb{S}_+^n$. Its parameters are accordingly $\boldsymbol{\theta} = \{\mathbf{P}_0, \dots, \mathbf{P}_L, \boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_L\}$, with $\boldsymbol{\xi}_l \in \mathcal{X}$. Due to a high number of parameters, it is also flexible, however the parameters $\boldsymbol{\theta}$ cannot be determined as efficiently as for SOS because (37) is biconvex as stated in the following.

Proposition 7. *Given the WSAQF Lyapunov candidate in (45) and a non-decreasing convex function $g(\cdot)$, the optimization (37) is a biconvex problem in $\boldsymbol{\theta}$ for fixed $\hat{\mathbf{x}}_*$.*

PROOF. The proof is analogous to Proposition 5.

This comparison shows, that sum of square is the most suitable, because it allows a high degree of flexibility (compared to the quadratic Lyapunov function) and a more efficient computation of its parameters than the WSAQF Lyapunov function.

5.3. Numerical considerations

From a numerical perspective, the optimization (37) achieves better results if the parameter set Θ includes a lower bound on the eigenvalues of the positive definite matrices $\mathbf{Q}, \mathbf{P}, \mathbf{P}_0, \dots, \mathbf{P}_L$ [22]. Otherwise the objective function is minimized by setting all entries close to zero, which is not the goal of the optimization (37). Therefore, we define the set

$$\mathbb{S}_\epsilon^m = \{ \mathbf{A} \in \mathbb{S}_+^m \mid \lambda_{\min}(\mathbf{A}) \geq \epsilon_\lambda \}, \quad m \in \mathbb{N} \quad (46)$$

and use it instead of \mathbb{S}_+^m to define the constraint set Θ . This will not affect the statements in Propositions 5, 6 and 7 because the constraint $\epsilon_\lambda - \lambda_{\min}(\mathbf{A}) < 0$ is convex (the eigenvalue operation is linear and the minimum of a linear functions is concave, its negative is convex).

For many practical applications, the equilibrium point of the unknown system (1) might be known a priori. In this case, the Lyapunov function search is simplified as the optimization is only over θ and therefore convex for $V_{\text{Sq}}(\cdot)$ and $V_{\text{SOS}}(\cdot)$.

6. NUMERICAL EVALUATION

6.1. Evaluation setup

For the numerical evaluation, we consider a programming by demonstration task, whose general goal is to mimic motion primitives from human experts

using a robotic system [10]. As demonstrations, we use recorded human hand-writing movements³ and use a dynamical system for the trajectory generation of the primitives. The dataset contains 24 goal-directed motions in two dimensions ($n = 2$). There are multiple (3 - 15) repetitions for each motion and all movements terminate at the same final location. Due to the generation by humans, it is difficult to design a suitable parametric model, and therefore, our data-driven nonparametric approach is well suited to model and generalize the given motions using a dynamical system. The common property of all motions is their convergence to a single goal point and to preserve this key feature, the dynamical system model must be stable. Therefore, our idea to stabilize a GPSSM is applicable here for this dataset.

We will assume that all trajectories are generated by a system of the form (1) which is asymptotically stable (Assumption 1) with the final location being the single equilibrium point $\mathbf{x}_* = \mathbf{0}$ ($N_* = 1$, $\mathcal{X} = \mathbb{R}^2$). The task is to identify a model of the form (7), which shows a high similarity to the true (but unknown) system (1) and also converges to (the estimated) equilibrium asymptotically (Objective 1). Additionally, we set up a probabilistic model (8), where the noise indicates the uncertainty for which we can also guarantee the convergence to an ultimate bound (Theorem 2) using the proposed techniques.

First, we focus on the estimation of an unknown equilibrium based on the optimization (37) for a single motion in Sec. 6.3. Second, in Sec. 6.4, we consider the equilibrium point to be known ($\mathbf{x}_* = \hat{\mathbf{x}}_* = \mathbf{0}$) to perform a quantitative comparison for the deterministic GP model on the full dataset (24 motions) among the different Lyapunov candidates. Finally, in Sec. 6.5 we will illustrate the almost sure ultimate boundedness in the probabilistic case from Theorem 2.

6.2. Implementation

Before presenting the results, we first provide an algorithmic overview of the proposed approach in Algorithm 1 and details regarding the implementation:⁴

³Available for download at <https://bitbucket.org/khansari/seds>

⁴Code for MATLAB is provided: <https://gitlab.lrz.de/ga68car/stablegps.git>

Algorithm 1 The proposed stable identification and prediction using GPSSMs.

- 1: take measurements \mathcal{D} at arbitrary locations (Assumption 3)
 - 2: optimize hyperparameters $\psi_j, \forall j$ according to (12)
 - 3: **if** $n\bar{\zeta} > l^2$ **then**
 - 4: take further measurements \mathcal{D}_{add} at specific locations (Assumption 6)
 - 5: **end if**
 - 6: select region of attraction i_* s.t. $\mathbf{x}_0 \in \mathcal{X}_{i_*}$
 - 7: optimize θ_{i_*} of the Lyapunov function $V_{\theta_{i_*}}^{i_*}(\cdot)$ and estimate the equilibrium $\hat{\mathbf{x}}_{i_*}$ using (37)
 - 8: initialize $\kappa = 0$ to utilize model in simulation
 - 9: **while** number of desired simulation steps not exceeded **do**
 - 10: **if** deterministic case **then**
 - 11: find optimal $\mathbf{u}(\mathbf{x}_\kappa)$ based on (17)
 - 12: $\mathbf{x}_{\kappa+1} = \boldsymbol{\mu}(\mathbf{x}_\kappa) + \mathbf{u}(\mathbf{x}_\kappa)$
 - 13: **else**
 - 14: find optimal $\mathbf{u}(\mathbf{x}_\kappa)$ based on (25)
 - 15: sample $\boldsymbol{\omega}_\kappa \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)$
 - 16: $\mathbf{x}_{\kappa+1} = \boldsymbol{\mu}(\mathbf{x}_\kappa) + \mathbf{u}(\mathbf{x}_\kappa) + \sqrt{\boldsymbol{\Sigma}(\mathbf{x}_\kappa)}\boldsymbol{\omega}_\kappa$
 - 17: **end if**
 - 18: $\kappa \leftarrow \kappa + 1$
 - 19: **end while**
-

As already mention, the measurements acquired in Line 1 are taken from the LASA dataset. In the first step of the proposed approach, the GPs are trained (line 2) based on the training data using a quasi-Newton method to perform the optimization (12). In the second major step, the optimal Lyapunov function based on (37) is determined using an interior-point method (Line 7). Because we want a fair comparison of all proposed Lyapunov candidates, we do not explicitly exploit the (bi) convexity with a well-suited algorithm. But, of course, finding the global optimum is simplified for the SOS and quadratic Lyapunov function. According to Propositions 5 and 6, we chose $\tilde{g}(\xi) = \xi$ as it results in a non-decreasing monotone function $g(\cdot)$.

For efficient implementation, we perform a Cholesky decomposition for all positive definite matrices and optimize over the nonzero entries of the lower triangular matrix. This avoids to explicitly impose the symmetry constraint. To enforce strict positive definiteness of the matrices $\mathbf{Q}, \mathbf{P}, \mathbf{P}_{0\dots L}$ their eigenvalues are enforced to be larger than 0.01 by adding this as a constraint to the numerical

optimization as stated in Sec. 5.3.

The simulation is initialized once at each starting point of the training trajectories and runs until the state reaches proximity of the (estimated) equilibrium point $\|\mathbf{x} - \hat{\mathbf{x}}_*\| < 5$ (or a limit of 1000 steps). The optimization (17) and (25) (in Line 11 and Line 14, respectively) are performed at each step by the inter-point algorithm, initialized with $\mathbf{u} = \hat{\mathbf{x}}_* - \boldsymbol{\mu}(\mathbf{x})$. The constraints (17b) (25b) are rewritten to

$$\begin{aligned} V(\boldsymbol{\mu}(\mathbf{x}_\kappa) + \mathbf{u}) - V(\mathbf{x}_\kappa) &\leq -\rho V(\mathbf{x}_\kappa) \\ \delta V(\mathbf{x}_\kappa) &\leq -\rho V(\mathbf{x}_\kappa) \end{aligned}$$

with $\rho = 0.02$ for the deterministic and probabilistic case, respectively to avoid the strict inequalities.

The Lyapunov functions presented in Sec. 5.2 are utilized as follows.

- A quadratic Lyapunov function $V_{\text{Sq}}(\cdot)$ defined in (41) with $n(n+1)/2 = 3$ free parameters (the elements of the Cholesky decomposition of \mathbf{P}).
- The SOS Lyapunov function defined in (44) with degree $2\mathcal{M} = 4$, resulting in $\mathcal{M}_n = 5$ monomials and therefore $\mathcal{M}_n(\mathcal{M}_n + 1)/2 = 15$ free parameters (Cholesky decomposition of \mathbf{Q}).
- The WSAQF Lyapunov function defined in (45) with $L = 3$ resulting in $(L + 1)n(n + 1)/2 + nL = 18$ free parameters for the Cholesky decompositions of $\mathbf{P}_{0\dots L}$ and $\boldsymbol{\xi}_{1\dots L}$.

6.3. Estimating the equilibrium

For this first example, we assume the true equilibrium point $\mathbf{x}_* = \mathbf{0}$ to be unknown and only consider the SOS Lyapunov function. Thus, for the optimization (37) a total of 17 (15 for SOS, 2 for the equilibrium point) variables are optimized. We utilized the motion called *Multi-Model 1* from the LASA dataset in this example.

The result for the simulation of the deterministic case (7) is shown Fig. 9. The equilibrium is identified to $\hat{\mathbf{x}}_* \approx [7.2 \ -12.5]^\top$, which is sufficiently close to the origin (considered as equilibrium in data generation) as the simulation is terminated for $\|\mathbf{x}_\kappa - \hat{\mathbf{x}}_*\| < 5$.

Unfortunately, there are currently no guarantees how far the estimated equilibria are from the true equilibria. However, the simulation suggested that for sufficiently many data points, which approach the equilibrium from different directions, the proposed approach leads to proper estimates. The theoretical investigations are left to future work.

It should also be considered, that the stabilization might lead to less smooth trajectories (compare Fig. 9 at $\mathbf{x} \approx [-60 \ 70]^\top$) which might be undesirable for human movement primitives. However, this varies from case to case and no general conclusion is possible.

6.4. Quantitative comparison

To enable a quantitative comparison of the Lyapunov functions presented in Sec. 5.2, we define the following measures for the precision of the learned model.

- The total area between the curve connecting the training points and the curve connecting the simulated trajectories.
- The average correction effort E_{cor} defined as $E_{\text{cor}} = \frac{\sum_{\kappa} \|\mathbf{u}_\kappa\|}{\sum_{\kappa} \|\mathbf{x}_\kappa\|}$ as sum over all steps and trajectories.

The results are shown in Table 1. Regarding flexibility, SOS outperforms the quadratic and WSAQF Lyapunov function on the employed dataset as shown in Table 1 which leads to higher precision in adherence to training data. Regarding the computational complexity, the search of the quadratic and the SOS Lyapunov function (37) are convex problems, and therefore have significant advantages, over the WSAQF. Computing the stabilizing command for the SOS has a disadvantage compared to WSAQF and quadratic Lyapunov functions as (17) is not convex. However, this drawback is inherent to flexible Lyapunov

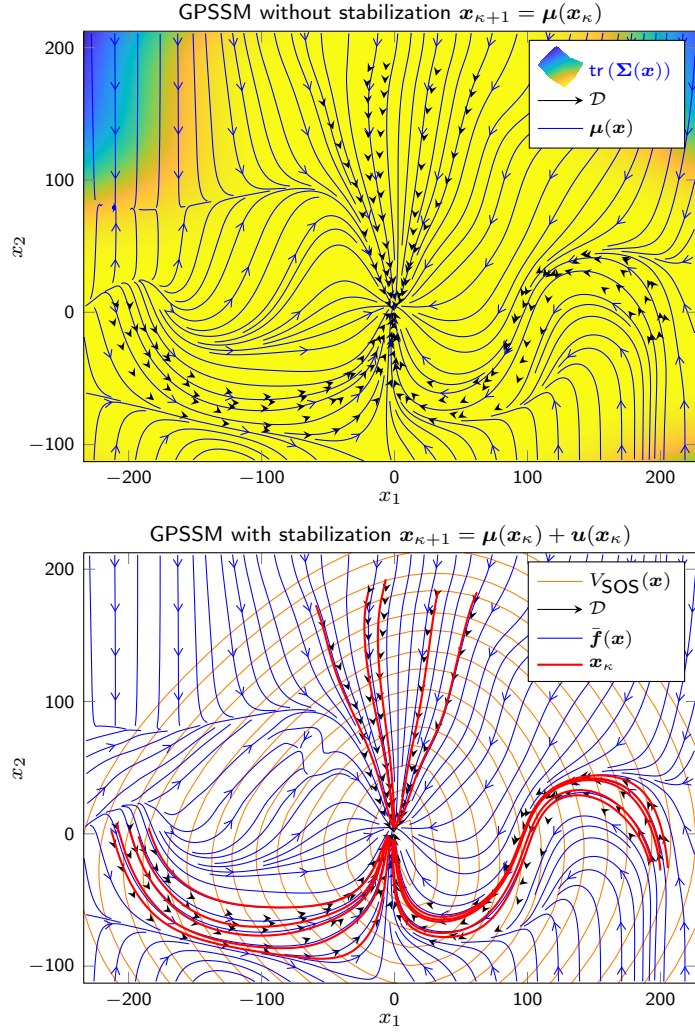


Figure 9: The non-stabilized GPSSM $\mathbf{x}_{\kappa+1} = \boldsymbol{\mu}(\mathbf{x}_{\kappa})$ (top) and the stabilized model $\mathbf{x}_{\kappa+1} = \boldsymbol{\mu}(\mathbf{x}_{\kappa}) + \mathbf{u}(\mathbf{x}_{\kappa})$ (bottom,) are visualized as streamlines (blue) along with the training data (black arrows) for the motion *Multi-Model 1*. The contour lines of the Lyapunov function are shown in orange. Without stabilizing command $\mathbf{u}(\cdot)$, there is a spurious attractor near $[-200 \ 80]^T$, which is eliminated through the stabilization. The trajectories (red), initialized at the starting state of each training trajectory, converge asymptotically to the identified equilibrium. The contour lines for the SOS control Lyapunov function are plotted in green.

$V(\mathbf{x}_\kappa)$	\min	E_{COR}	\max	\min	Area Error	\max	(37) convex	(17) convex
Quadratic	0	0.224	1.1	165	$8.1 \cdot 10^3$	$3.5 \cdot 10^4$	yes	yes
SOS	0	0.038	0.46	103	$3.3 \cdot 10^3$	$1.2 \cdot 10^4$	yes	no
WSAQF	0	0.074	0.64	103	$4.0 \cdot 10^3$	$1.7 \cdot 10^4$	no	yes

Table 1: [Average, minimum and maximum performance](#) of quadratic, SOS and WSAQF Lyapunov functions across the different motions in the LASA handwriting dataset. The convexity refers to the case for which \mathbf{x}_* is known.

candidates: If only convex Lyapunov candidates are permitted, more complex systems cannot be captured.

6.5. Probabilistic simulation

To evaluate the probabilistic setting (8), we simulate as follows.

- GPSSM without stabilization: Realizations of the trajectories are generated by drawing in each step from

$$\mathbf{x}_{\kappa+1} \sim \mathcal{N}(\boldsymbol{\mu}(\mathbf{x}_\kappa), \boldsymbol{\Sigma}(\mathbf{x}_\kappa)), \quad (47)$$

using the definitions in (16).

- GPSSM with stabilization through the SOS Lyapunov function: At each step $\mathbf{u}(\cdot)$ is computed according to (25) and the next step is drawn from

$$\mathbf{x}_{\kappa+1} \sim \mathcal{N}(\boldsymbol{\mu}(\mathbf{x}_\kappa) + \mathbf{u}(\mathbf{x}_\kappa), \boldsymbol{\Sigma}(\mathbf{x}_\kappa)), \quad (48)$$

which is equivalent to (8). The stabilizing command is independent of the realization of $\mathbf{x}_{\kappa+1}$ which is unknown when \mathbf{u} is computed. It only depends on $\boldsymbol{\mu}(\mathbf{x}_\kappa), \boldsymbol{\Sigma}(\mathbf{x}_\kappa)$.

For both cases, the system is initialized twice with $\mathbf{x}_0 = [-150 \ -120]^\top$ as shown in Fig. 10. For the GPSSM without stabilization both trajectories do not converge in contrast to the trajectories of the stabilized model. Based on Theorem 1 only ultimate boundedness can be guaranteed because $n\bar{\zeta} > \bar{l}^2$ for the hyperparameters obtain from the optimization (12).

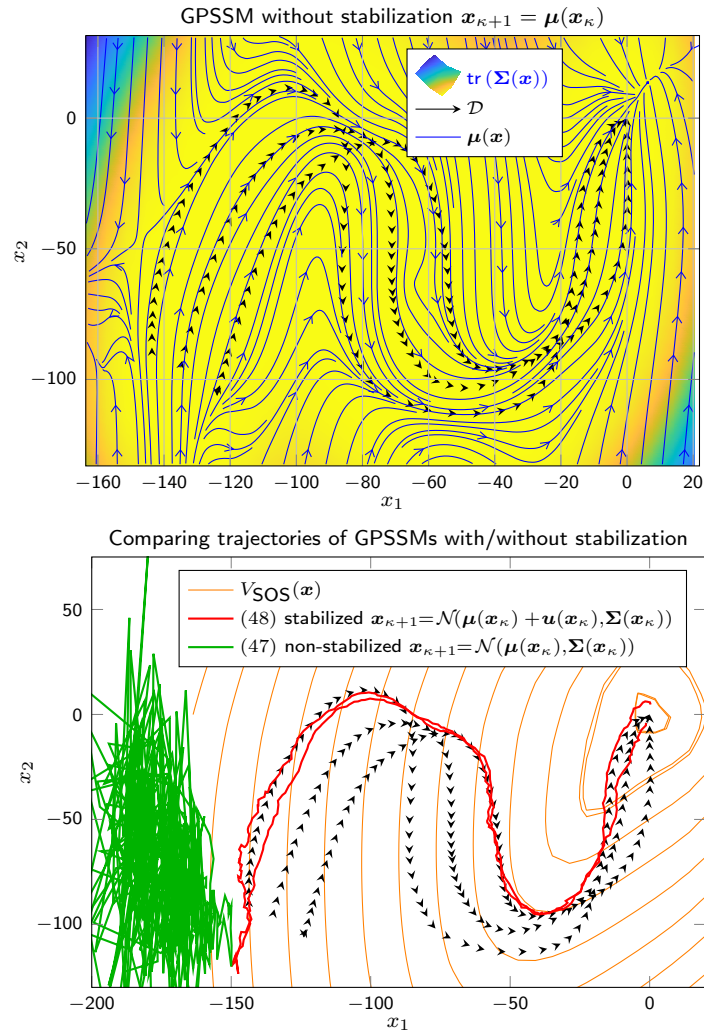


Figure 10: The GPSSM without stabilization (top) and the stochastic simulations (bottom) along with the training data (black arrows) for the *N-Shape* motion. The contour lines of the Lyapunov function are shown in orange. The trajectories (red with stabilization $\mathbf{u}(\cdot)$, green without stabilization $\mathbf{u}(\cdot)$) are initialized twice at $[-150 \ -120]^T$.

7. CONCLUSION

In this article, we propose a novel approach for learning stable Gaussian process state-space models with multiple equilibria using control Lyapunov functions. For the augmented GP model (5) we show, that the deterministic interpretation allows to model arbitrarily complex continuous functions $\mathbf{f}(\cdot)$ consistently and that for infinite data, the model approaches the true function (for a properly chosen control Lyapunov function) as derived in Proposition 1. The resulting model is uniformly globally asymptotically stable according to Theorem 1. In the probabilistic case, we derive conditions for almost sure uniform asymptotic stability and uniform ultimate boundedness of the model and compute the ultimate bound (Theorem 2). Furthermore, we derive the number of additional training points required to guarantee asymptotic stability almost surely in Theorem 3. These results are derived for the squared exponential kernel only, because it is the most commonly employed kernel for GP regression. However, Theorem 1 and Corollary 1 can directly be extended to arbitrary kernel choices. We leave it to future work to extend the results on Theorem 2 and Theorem 3 to other classes of kernels.

Additionally, we show how parametric Lyapunov functions are efficiently learned from data. The approach allows the estimation of a finite number of stable equilibria (as long as their domain of attraction is provided). We show that the required optimization is (bi-) convex for particular choices of Lyapunov candidates and that the GPSSM is only minimal distorted. For future work, it is desired to utilize nonparametric Lyapunov function to allow a pure data-driven model. Furthermore, it is important to provide theoretical error bounds on how far the estimated equilibria are from the true equilibria and to estimate the regions of attraction from data. In the simulation, we show that the proposed framework allows precise identification based on a real world dataset.

Acknowledgments

Funding: This work was supported by the European Research Council (ERC) Starting Grant "Control based on Human Models (conhumo)" under grant agreement number 337654.

References

- [1] Christian Agrell. Gaussian processes with linear operator inequality constraints. *Journal of Machine Learning Research*, 20(135):1–36, 2019.
- [2] Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.
- [3] Harry H. Asada, Faye Wu, Alexandre Girard, and Michaelle Mayalu. A data-driven approach to precise linearization of nonlinear dynamical systems in augmented latent space. In *American Control Conference (ACC)*, pages 7–1844. IEEE, July 2016.
- [4] Thomas Beckers and Sandra Hirche. Equilibrium distributions and stability analysis of Gaussian process state space models. In *Conference on Decision and Control (CDC)*, pages 6355–6361. IEEE, 2016.
- [5] Thomas Beckers and Sandra Hirche. Stability of Gaussian process state space models. In *European Control Conference (ECC)*, pages 2275–2281, 2016.
- [6] Thomas Beckers, Dana Kulić, and Sandra Hirche. Stable Gaussian process based tracking control of Euler-Lagrange systems. *Automatica*, 23(103):390–397, 2019.
- [7] Thomas Beckers, Jonas Umlauft, and Sandra Hirche. Mean square prediction error of misspecified Gaussian process models. In *Conference on Decision and Control (CDC)*, pages 1162–1167, December 2018.

- [8] Felix Berkenkamp, Riccardo Moriconi, Angela Schoellig, and Andreas Krause. Safe learning of regions of attraction for uncertain, nonlinear systems with Gaussian processes. *arXiv preprint arXiv:1603.04915*, 2016.
- [9] Felix Berkenkamp and Angela P. Schoellig. Safe and robust learning control with Gaussian processes. In *European Control Conference (ECC)*, pages 2496–2501, July 2015.
- [10] Aude Billard, Sylvain Calinon, Rüdiger Dillmann, and Stefan Schaal. *Robot programming by demonstration*. Springer, 2008.
- [11] Caroline Blocher, Matteo Saveriano, and Dongheui Lee. Learning stable dynamical systems using contraction theory. In *International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 124–129. IEEE, June 2017.
- [12] Byron Boots, Geoffrey J. Gordon, and Sajid M. Siddiqi. A constraint generation approach to learning stable linear dynamical systems. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1329–1336. Curran Associates, Inc., 2008.
- [13] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- [14] Carl E. Deisenroth, Marc P. and Rasmussen and Jan Peters. Gaussian process dynamic programming. *Neurocomputing*, 72(7-9):1508–1524, March 2009.
- [15] Marc P. Deisenroth. *Efficient reinforcement learning using Gaussian processes*, volume 9. KIT Scientific Publishing, 2010.
- [16] Heinz Werner Engl, Martin Hanke, and Andreas Neubauer. *Regularization of inverse problems*, volume 375. Springer Science & Business Media, 1996.
- [17] Wouter Favoreel, Bart De Moor, and Peter Van Overschee. Subspace state space system identification for industrial processes. *Journal of process control*, 10(2):149–155, 2000.

- [18] Roger Frigola-Alcade. *Bayesian time series learning with Gaussian processes*. PhD thesis, University of Cambridge, 2015.
- [19] Motonobu Kanagawa, Philipp Hennig, Dino Sejdinovic, and Bharath K. Sriperumbudur. Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv preprint arXiv:1807.02582*, 2018.
- [20] Hassan K. Khalil and Jessy. W. Grizzle. *Nonlinear systems*, volume 3. Prentice hall New Jersey, 1996.
- [21] Seyed Mohammad Khansari-Zadeh and Aude Billard. Learning stable nonlinear dynamical systems with Gaussian mixture models. *IEEE Transactions on Robotics (T-RO)*, 27(5):943–957, 2011.
- [22] Seyed Mohammad Khansari-Zadeh and Aude Billard. Learning control Lyapunov function to ensure stability of dynamical system-based robot reaching motions. *Robotics and Autonomous Systems*, 62(6):752–765, 2014.
- [23] Andreas Kirsch. *An introduction to the mathematical theory of inverse problems*, volume 120. Springer Science & Business Media, 2011.
- [24] Jus Kocijan. *Modelling and Control of Dynamic Systems Using Gaussian Process Models*. Springer, 2016.
- [25] Jus. Kocijan, Agathe. Girard, Blaz Banko, and Roderick Murray-Smith. Dynamic systems identification with Gaussian processes. *Mathematical and Computer Modelling of Dynamical Systems*, 11(4), 411-424, 2005.
- [26] Harold Joseph Kushner. *Introduction to stochastic control*. Holt, Rinehart and Winston New York, 1971.
- [27] Armin Lederer, Jonas Umlauft, and Sandra Hirche. Posterior variance analysis of Gaussian processes. *arXiv preprint: arXiv:1906.01404*, June 2019.
- [28] Yan Li, Weihai Zhang, and Xikui Liu. Stability of nonlinear stochastic discrete-time systems. *Journal of Applied Mathematics*, 2013, 2013.

- [29] Miao Liu, Girish Chowdhary, Bruno Castra da Silva, Shih Y. Liu, and Jonathan P. How. Gaussian processes for learning and control: A tutorial with examples. *IEEE Control Systems Magazine*, 38(5):53–86, October 2018.
- [30] Lennart Ljung. *System Identification*. Prentice Hall PTR, NJ, USA, 1998.
- [31] Jan M. Maciejowski. Guaranteed stability with subspace methods. *Systems & Control Letters*, 26(2):153–156, 1995.
- [32] Andrew James McHutchon. *Nonlinear modelling and control using Gaussian processes*. PhD thesis, University of Cambridge, 2015.
- [33] Jose R. Medina and Aude Billard. Learning stable task sequences from demonstration with linear parameter varying systems and hidden markov models. In *Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 175–184. PMLR, November 2017.
- [34] Jose R. Medina, Lee Dongheui, and Sandra Hirche. Risk-sensitive optimal feedback control for haptic assistance. In *International Conference on Robotics and Automation (ICRA)*, pages 1025–1031. IEEE, May 2012.
- [35] Oliver Nelles. *Nonlinear system identification: From classical approaches to neural networks and fuzzy models*. Springer Science & Business Media, 2013.
- [36] Klaus Neumann and Jochen J. Steil. Learning robot motions with stable dynamical systems under diffeomorphic transformations. *Robotics and Autonomous Systems*, 70:1–15, 2015.
- [37] Yunpeng Pan and Evangelos Theodorou. Probabilistic differential dynamic programming. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1907–1915. Curran Associates, Inc., 2014.
- [38] Antonis Papachristodoulou and Stephen Prajna. On the construction of Lyapunov functions using the sum of squares decomposition. In *Confer-*

- ence on Decision and Control (CDC), volume 3, pages 3482–3487. IEEE, December 2002.
- [39] Pablo A. Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD thesis, California Institute of Technology, 2000.
- [40] David L. Phillips. A technique for the numerical solution of certain integral equations of the first kind. *Journal of the ACM (JACM)*, 9(1):84–97, 1962.
- [41] Gianluigi Pillonetto and Giuseppe De Nicolao. A new kernel-based approach for linear system identification. *Automatica*, 46(1):81–93, 2010.
- [42] Gianluigi Pillonetto, Francesco Dinuzzo, Tianshi Chen, Giuseppe De Nicolao, and Lennart Ljung. Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3):657–682, 2014.
- [43] Lukas Pöhler, Jonas Umlauf, and Sandra Hirche. Uncertainty-based human trajectory tracking with stable Gaussian process state space models. In *IFAC Conference on Cyber-Physical & Human Systems (CPHS)*, Miami, December 2018. IFAC.
- [44] Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, USA, January 2006.
- [45] Saburo Saitoh and Yoshihiro Sawano. *Theory of reproducing kernels and applications*. Springer, 2016.
- [46] Stefan Schaal, Jan Peters, Jun Nakanishi, and Auke Ijspeert. Learning movement primitives. In *The Eleventh International Symposium Robotics Research.*, pages 561–572. Springer, 2005.
- [47] Robert Schaback. Improved error bounds for scattered data interpolation by radial basis functions. *Mathematics of Computation*, pages 201–216, 1999.

- [48] Matthias W. Seeger, Sham M. Kakade, and Dean. P. Foster. Information consistency of nonparametric Gaussian process methods. *IEEE Transactions on Information Theory*, 54(5):2376–2382, May 2008.
- [49] Stefano Soatto, Gianfranco Doretto, and Ying Nian Wu. Dynamic textures. In *International Conference on Computer Vision (CCV)*, volume 2, pages 439–446. IEEE, 2001.
- [50] 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, May 2012.
- [51] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: An introduction*. MIT Press, Cambridge, MA, USA, 1998.
- [52] Vladimir Temlyakov. A remark on covering. *arXiv preprint arXiv:1301.3043*, 2013.
- [53] Evangelos Theodorou, Jonas Buchli, and Stefan Schaal. A generalized path integral control approach to reinforcement learning. *Journal of Machine Learning Research*, 11(Nov):3137–3181, 2010.
- [54] Kostas Triantafyllopoulos. Moments and cumulants of the multivariate real and complex Gaussian distributions. University Bristol, 2002.
- [55] Jonas Umlauft, Thomas Beckers, and Sandra Hirche. A scenario-based optimal control approach for Gaussian process state space models. In *European Control Conference (ECC)*, pages 1386–1392, 6 2018.
- [56] Jonas Umlauft, Thomas Beckers, Melanie Kimmel, and Sandra Hirche. Feedback linearization using Gaussian processes. In *Conference on Decision and Control (CDC)*, pages 5249–5255. IEEE, December 2017.
- [57] Jonas Umlauft, Armin Lederer, and Sandra Hirche. Learning stable Gaussian process state space models. In *American Control Conference (ACC)*, pages 1499–1504. IEEE, IEEE, May 2017.

- [58] 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, July 2018.
- [59] Peter Van Overschee and B. L. De Moor. *Subspace identification for linear systems: Theory—Implementation—Applications*. Springer Science & Business Media, 2012.
- [60] Jack M. Wang, David J. Fleet, and Aaron Hertzmann. Gaussian process dynamical models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1441–1448, 2005.
- [61] Christopher K. I. Williams and Francesco Vivarelli. Upper and lower bounds on the learning curve for Gaussian processes. *Machine Learning*, 40(1):77–102, 2000.
- [62] Wolfram|Alpha. Solution based on the analytic continuation of the product log function. Online, January 2019.
- [63] David H. Wolpert. The supervised learning no-free-lunch theorems. In *Soft Computing and Industry*, pages 25–42. Springer, 2002.