

Local Asymptotic Stability Analysis and Region of Attraction Estimation with Gaussian Processes*

Armin Lederer and Sandra Hirche

Abstract—Determining the region of attraction of nonlinear systems is a difficult problem, which is typically approached by means of Lyapunov theory. State of the art approaches either provide high flexibility regarding the Lyapunov function or parallelizability of computation. Aiming at both, flexibility and parallelizability, we propose a method to obtain a Lyapunov-like function for stability analysis by learning the infinite horizon cost function with a Gaussian process based on approximate dynamic programming. We develop a novel approach to characterize the region of attraction using a Lyapunov-like function, which is analyzed with a sampling-based interval analysis algorithm. Since each interval can be examined independently, the algorithm allows both parallelizable analysis and flexible construction of the Lyapunov-like function.

I. INTRODUCTION

A fundamental problem of engineering lies in proving the satisfaction of performance bounds and safety constraints. These problems can be approached with the help of dynamical systems theory by guaranteeing local asymptotic stability on a suitable subset of the state space, called region of attraction (ROA). This approach is used in a variety of control applications, such as aviation [1] and power systems [2]. Furthermore, it has also been used in other fields, such as medicine [3] and ecology [4].

Several methods exist to analyze local asymptotic stability and underestimate ROAs for nonlinear discrete-time systems with special structure, such as polynomial recurrent dynamics [5]. However, these approaches are inherently limited in their applicability to a restricted class of systems. Other approaches construct sum of squares Lyapunov functions, e.g., by solving a simulation-based linear program [6] or through convex optimization [7]. Yet the restriction to sum of squares Lyapunov functions reduces the flexibility of this approach, such that it might be impossible to obtain the true ROA. An approach which is capable of approximating the region of attraction of exponentially stable equilibria is presented in [8]. It is based on a discretization of the state space, which is exploited to generate a piecewise continuous affine Lyapunov function by solving a linear program. Triangulation guarantees that the Lyapunov conditions are satisfied inside the continuous intervals. Finer grids allow to increase the quality of the ROA estimate. Yet it does not scale well to high dimensional problems or systems requiring very fine grids due to

the centralized nature of the linear optimization. An approach which allows efficient parallelization is presented in [9]. It uses the accumulated cost along trajectories of specified length as a Lyapunov candidate, and proves the satisfaction of the Lyapunov conditions in a sampling-based fashion with the help of interval analysis [10]. However, this approach can exhibit high computational cost for strongly nonlinear systems that require long trajectories for a good Lyapunov candidate. Therefore, there is a lack of methods which allow the parallelizable stability analysis of highly nonlinear systems.

The contribution of this paper is a novel approach to estimate the ROA of nonlinear systems in a flexible and parallelizable way exploiting Gaussian process (GP) regression to learn the infinite horizon cost function, which can be used as Lyapunov function for stable systems [11]. The infinite horizon cost is learned efficiently with a Gaussian process by exploiting the Bellman equation [12]. Since the learned cost might violate the Lyapunov conditions around the origin due to regression errors, we derive a theorem allowing to extend known regions of attraction through a Lyapunov-like function. Using interval analysis tailored to Gaussian processes, the learned infinite horizon cost is analyzed in a sampling-based fashion to determine the extended ROA. Thereby, our proposed algorithm provides high flexibility for the ROA and efficient analysis through parallelized interval analysis.

The remainder of this paper is structured as follows: after a review of Gaussian process regression in Section II, we propose a method for the computation of the ROA in Section III, which is evaluated on a nonlinear system in Section IV.

II. GAUSSIAN PROCESS REGRESSION

Gaussian processes are a supervised machine learning method for the regression of nonlinear functions. A GP is a stochastic process, which assigns a joint Gaussian distribution to any finite subset¹ $\{\mathbf{x}_i, \dots, \mathbf{x}_N\} \subset \mathbb{X}$ from a continuous input domain $\mathbb{X} \subset \mathbb{R}^d$ [13]. A GP is completely specified by its mean function $m(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}$ and covariance kernel $k(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$. Approximate models can be incorporated into GP regression through the mean function $m(\cdot)$. Without such prior knowledge, the mean function $m(\cdot)$ is typically set to zero, which is assumed in the following without loss of generality. The covariance kernel is used to encode abstract prior information about the regressed function, such as the smoothness or periodicity. Probably the most

*This work was supported by the European Research Council Starting Grant "Control based on Human Models" under Grant 337654. A. L. gratefully acknowledges financial support from the German Academic Scholarship Foundation. Published under ©2019 IEEE.

Both authors are with the Department of Electrical and Computer Engineering, Technical University of Munich, 80333 Munich, Germany [armin.lederer, hirche]@tum.de

¹Notation: Lower/upper case bold symbols denote vectors/matrices, $\mathbb{R}_+/\mathbb{N}_+$ all real/integer positive numbers, \mathbf{I}_n the $n \times n$ identity matrix, $\|\cdot\|$ the Euclidean norm and $\langle \cdot, \cdot \rangle$ the scalar product.

commonly used kernel is the squared exponential kernel [13]

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\sum_{i=1}^d \frac{(x_i - x'_i)^2}{2l_i^2}\right), \quad (1)$$

where l_i are the characteristic length scales of the kernel. An important property of the squared exponential kernel is the fact that it is universal [14], i.e., a GP with this kernel can approximate any continuous function arbitrarily well.

For regression with a GP, we assume that the training data $\mathbb{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^N$ is generated by a function $f(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}$ through

$$y^{(i)} = f(\mathbf{x}^{(i)}) + \omega^{(i)}, \quad (2)$$

where $\omega^{(i)} \sim \mathcal{N}(0, \sigma_n^2)$ are i.i.d. random variables. If noiseless training data is provided, i.e., $\sigma_n^2 = 0$, this is considered a noiseless regression problem. In the following, we consider the more general noisy regression problem. By defining the data covariance matrix \mathbf{K} and the covariance vector $\mathbf{k}(\mathbf{x}^*)$ through $K_{ij} = k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ and $k_i(\mathbf{x}^*) = k(\mathbf{x}^{(i)}, \mathbf{x}^*)$, the predictive mean can be derived as

$$\mu(\mathbf{x}^*) = \mathbb{E}[f(\mathbf{x}^*) | \mathbb{D}, \mathbf{x}^*] = \mathbf{k}^T(\mathbf{x}^*)\boldsymbol{\lambda}, \quad (3)$$

with $\boldsymbol{\lambda} = (\mathbf{K} + \sigma_n^2 \mathbf{I}_N)^{-1} \mathbf{y}$ and target vector $\mathbf{y} = [y^{(1)} \dots y^{(N)}]^T$. The characteristic length scales l_i of the covariance kernel $k(\cdot, \cdot)$ and the observation noise variance σ_n^2 are considered hyperparameters of the GP. They are typically determined by maximization of the log-likelihood according to Bayesian principles [13].

III. LEARNING AND ANALYZING LYAPUNOV CANDIDATES

A major difficulty in Lyapunov-based stability analysis is the choice of suitable Lyapunov candidates. For small neighborhoods of the equilibrium, this problem is typically easy to solve, e.g., by quadratic Lyapunov functions. We derive a theorem that allows to extend this ROA using Lyapunov-like functions in Section III-A. In Section III-B, we propose a method to generate a suitable Lyapunov-like function by formulating the approximation of the infinite horizon cost as a regression problem. Finally, in Section III-C, we introduce a sampling-based analysis of the Lyapunov-like function, which uses interval analysis specifically tailored to GPs.

A. Lyapunov Theory based Asymptotic Stability Analysis

Stability of systems is a property that is defined for trajectories of infinite length. For this reason, we recursively define trajectories of a discrete-time dynamical system by

$$\mathbf{x}_0 = \mathbf{x} \quad (4a)$$

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) \quad \forall k \in \mathbb{N}. \quad (4b)$$

Then, stability is defined in the following way [15].

Definition 1: A dynamical system $\mathbf{f}(\cdot)$ is called locally stable if, for all $\epsilon > 0$, there exists a $\delta > 0$ such that $\|\mathbf{x}_0\| \leq \delta$ implies $\|\mathbf{x}_k\| \leq \epsilon$, $\forall k \in \mathbb{N}$. If it is additionally attractive on a set $\mathbb{A} \subset \mathbb{R}^d$, i.e., $\lim_{k \rightarrow \infty} \|\mathbf{x}_k\| = 0$, $\forall \mathbf{x}_0 \in \mathbb{A}$, then it is asymptotically stable on the ROA \mathbb{A} .

Note that this form of stability is slightly weaker than the frequently considered \mathcal{KL} -stability [16]. This is due to the fact that it allows stable systems to move arbitrarily far away from the equilibrium before convergence.

Analyzing the ROA of nonlinear dynamical systems is typically relaxed to finding level sets of Lyapunov functions. Although automatically finding a suitable Lyapunov function is generally a hard problem in itself, Lyapunov functions for small neighborhoods \mathbb{S} can usually be determined rather simply, e.g., using quadratic functions or sum of squares methods [7]. Therefore, we will provide a novel theorem that allows to extend a known stable neighborhood of the origin using a second function, which has to satisfy far weaker conditions than a Lyapunov function.

Theorem 1: Consider a dynamical system $\mathbf{f}(\cdot)$, which is known to be asymptotically stable in the set \mathbb{S} with $\mathbf{0} \in \text{int}(\mathbb{S})$, and a bounded function $V : \mathbb{R}^d \rightarrow \mathbb{R}$, $|V(\mathbf{x})| \leq \bar{v}$, $\forall \mathbf{x} \in \mathbb{R}^d$. Choose $\epsilon \in \mathbb{R}_+$ and define a set

$$\mathbb{D}_\epsilon \subseteq \{\mathbf{x} \in \mathbb{R}^d : V(\mathbf{f}(\mathbf{x})) - V(\mathbf{x}) \leq -\epsilon\}, \quad (5)$$

such that there exist states $\mathbf{x} \in \mathbb{D}_\epsilon$ with $\mathbf{f}(\mathbf{x}) \notin \mathbb{D}_\epsilon \cup \mathbb{S}$. Let c be defined as

$$c = \inf_{\mathbf{x} \in \{\mathbf{x} \in \mathbb{D}_\epsilon : \mathbf{f}(\mathbf{x}) \notin \mathbb{D}_\epsilon \cup \mathbb{S}\}} V(\mathbf{f}(\mathbf{x})). \quad (6)$$

Then, the dynamical system is asymptotically stable on the set $\mathbb{V} = \{\mathbf{x} \in \mathbb{D}_\epsilon : V(\mathbf{x}) < c\} \cup \mathbb{S}$.

Proof: First, note that asymptotic stability in the ball \mathbb{S} implies local stability of the system. In order to show the attraction to the origin, we first prove forward invariance of \mathbb{V} . Forward invariance on \mathbb{S} trivially follows from asymptotic stability. For every $\mathbf{x} \in \mathbb{V} \setminus \mathbb{S}$ with $V(\mathbf{x}) \leq c$, we have $V(\mathbf{f}(\mathbf{x})) \leq c - \epsilon$ since $\mathbb{V} \setminus \mathbb{S} \subset \mathbb{D}_\epsilon$. Therefore, we obtain $\mathbf{f}(\mathbf{x}) \in \mathbb{V}$ due to (6) and, consequently, \mathbb{V} is a forward invariant set of $\mathbf{f}(\cdot)$. Based on the forward invariance of \mathbb{V} , we prove convergence to the asymptotically stable set \mathbb{S} by contradiction. For this reason, assume that there exists an $\mathbf{x}_0 \in \mathbb{V} \setminus \mathbb{S}$, such that $\mathbf{x}_k \in \mathbb{V} \setminus \mathbb{S}$, $\forall k \in \mathbb{N}$, where \mathbf{x}_k is recursively defined by (4). This implies

$$V(\mathbf{x}_k) \leq V(\mathbf{x}_0) - k\epsilon \quad (7)$$

for all $k \in \mathbb{N}$ due to forward invariance of \mathbb{V} . However, due to the upper bound of $V(\cdot)$, there exists a \bar{k} such that

$$V(\mathbf{x}_k) < -\bar{v} \quad \forall k \geq \bar{k}, \quad (8)$$

which violates the assumption $V(\mathbf{x}) \geq -\bar{v}$, $\forall \mathbf{x} \in \mathbb{R}^d$. Hence, each trajectory starting in \mathbb{V} must eventually enter the asymptotically stable set \mathbb{S} , which implies asymptotic stability on \mathbb{V} . ■

Note that this theorem does not require the knowledge of a forward invariant set of $\mathbf{f}(\cdot)$, which is an assumption often required in similar stability theorems. This is a major advantage because determining an invariant set is a difficult problem in itself. Even though \mathbb{D}_ϵ is required to be not forward invariant for the infimum in (6) to be defined, this is not restrictive because forward invariance of sets can be destroyed by removing points. Therefore, one can remove

points from \mathbb{D}_ϵ until the infimum in (6) is well defined, hence it is not necessary to know a forward invariant set of $\mathbf{f}(\cdot)$.

Furthermore, note that the conditions on the function $V(\cdot)$ are significantly weaker than those for Lyapunov functions, since only boundedness is required explicitly. Nevertheless, in order to extend \mathbb{S} , $V(\cdot)$ has to fulfill another property that makes it a Lyapunov-like function.

Lemma 1: A Lyapunov-like function $V(\cdot)$ can extend the known ROA \mathbb{S} only if its minimizing state

$$\mathbf{x}^* = \underset{\mathbf{x} \in \{\mathbf{x} \in \mathbb{R}^d: \exists \mathbf{x}' \in \mathbb{D}_\epsilon: \mathbf{x} = \mathbf{f}(\mathbf{x}')\}}{\text{arg min}} V(\mathbf{x}) \quad (9)$$

lies in \mathbb{S} .

Proof: We prove this lemma by contradiction. For this reason, assume that $\mathbf{x}^* \notin \mathbb{S}$ and \mathbb{S} is a strict subset of \mathbb{V} . Since $V(\mathbf{x}^*)$ is the minimum of $V(\cdot)$, the decrease condition cannot be satisfied at \mathbf{x}^* , i.e., $\mathbf{x}^* \notin \mathbb{D}_\epsilon$. Hence, it follows from (6) that $c = V(\mathbf{x}^*)$. However, this implies $\mathbb{S} = \mathbb{V}$ due to minimality of $V(\mathbf{x}^*)$, which contradicts our initial assumption $\mathbb{S} \subset \mathbb{V}$. Therefore, $V(\cdot)$ can only extend \mathbb{S} if $\mathbf{x}^* \in \mathbb{S}$. ■ Lemma 1 can be seen as a weaker version of the positive definiteness requirement for Lyapunov functions, which essentially requires the minimum of a function to lie at the origin.

B. Learning the Infinite Horizon Cost

Since the Lyapunov-like function in Theorem 1 has to satisfy properties similar to a Lyapunov function, a suitable approach to obtain it is based on approximating a Lyapunov function. Given an immediate cost $l(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}_+$, which is typically chosen to be monotonically increasing with growing $\|\mathbf{x}\|$, it is well known that the infinite horizon cost

$$V(\mathbf{x}) = \sum_{k=0}^{\infty} \gamma^k l(\mathbf{x}_k), \quad (10)$$

with the discount factor γ and the trajectory \mathbf{x}_k defined in (4), can be a suitable Lyapunov function for stable nonlinear systems [11]. In general this function cannot be calculated analytically due to the infinite summation. However, it is clear that this cost function satisfies the Bellman equation

$$V(\mathbf{x}) - (l(\mathbf{x}) + \gamma V(\mathbf{f}(\mathbf{x}))) = 0 \quad \forall \mathbf{x} \in \mathbb{R}^d. \quad (11)$$

Therefore, we propose a method which allows to learn an approximation of the infinite horizon cost function satisfying the Bellman equation at finitely many sample points.

Theorem 2: Consider the nonlinear dynamics $\mathbf{f}(\cdot)$, an immediate cost function $l(\cdot)$, a discount factor γ , and an universal kernel $k(\cdot, \cdot)$. An approximation $\tilde{V}(\cdot)$ of the system's infinite horizon cost function satisfying the Bellman equation at $N \in \mathbb{N}$ sampling points $\mathbf{x}^{(i)}$, $i = 1, \dots, N$, can be obtained through noiseless GP regression with the modified kernel

$$\tilde{k}(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - \gamma k(\mathbf{x}, \mathbf{f}(\mathbf{x})) - \gamma k(\mathbf{f}(\mathbf{x}), \mathbf{x}) + \gamma^2 k(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{x})). \quad (12)$$

The resulting approximation $\tilde{V}(\cdot)$ is given by

$$\tilde{V}(\cdot) = \sum_{i=1}^N \tilde{\lambda}_i \left(k(\mathbf{x}^{(i)}, \cdot) - \gamma k(\mathbf{f}(\mathbf{x}^{(i)}), \cdot) \right), \quad (13)$$

where

$$\tilde{\lambda} = \tilde{\mathbf{K}}^{-1} [l(\mathbf{x}^{(1)}) \dots l(\mathbf{x}^{(N)})]^T. \quad (14)$$

Proof: Due to Mercer's theorem [17], there exists a feature map $\phi(\cdot) : \mathbb{R}^d \rightarrow \mathbb{F} \subset \mathbb{R}^\infty$ for every kernel $k(\cdot, \cdot)$, such that

$$k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle. \quad (15)$$

Feature maps have the advantage that they allow to perform nonlinear regression using linear methods in a feature space \mathbb{F} . Due to the universality of $k(\cdot, \cdot)$, there exists a parameter vector $\boldsymbol{\theta}$ such that

$$V(\mathbf{x}^{(i)}) = \langle \boldsymbol{\theta}, \phi(\mathbf{x}^{(i)}) \rangle \quad \forall i = 1, \dots, N \quad (16)$$

$$V(\mathbf{f}(\mathbf{x}^{(i)})) = \langle \boldsymbol{\theta}, \phi(\mathbf{f}(\mathbf{x}^{(i)})) \rangle \quad \forall i = 1, \dots, N. \quad (17)$$

Using this parameterization of the cost function in the Bellman equation (11) yields

$$0 = \langle \boldsymbol{\theta}, \phi(\cdot) \rangle - (l(\cdot) + \gamma \langle \boldsymbol{\theta}, \phi(\mathbf{f}(\cdot)) \rangle) = -l(\cdot) + \langle \boldsymbol{\theta}, \boldsymbol{\psi}(\cdot) \rangle, \quad (18)$$

where the modified feature map $\boldsymbol{\psi}(\cdot)$ is defined as

$$\boldsymbol{\psi}(\cdot) = \phi(\cdot) - \gamma \phi(\mathbf{f}(\cdot)) \quad (19)$$

due to linearity of the scalar product. Therefore, finding a function $\tilde{V}(\cdot)$, which satisfies the Bellman equation at the sampling points, reduces to finding a parameter vector $\boldsymbol{\theta}$ such that

$$l(\mathbf{x}^{(i)}) = \langle \boldsymbol{\theta}, \boldsymbol{\psi}(\mathbf{x}^{(i)}) \rangle \quad \forall i = 1, \dots, N. \quad (20)$$

This problem can be solved with GP regression since $\boldsymbol{\psi}(\cdot)$ is merely a linear combination of feature maps, hence it also defines a feature map $\tilde{k}(\cdot, \cdot) = \langle \boldsymbol{\psi}(\cdot), \boldsymbol{\psi}(\cdot) \rangle$. This modified kernel can be expressed in terms of the original kernel as

$$\tilde{k}(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - \gamma k(\mathbf{x}, \mathbf{f}(\mathbf{x})) - \gamma k(\mathbf{f}(\mathbf{x}), \mathbf{x}) + \gamma^2 k(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{x})). \quad (21)$$

Regressing the training data set $\mathbb{D}_N \{(\mathbf{x}^{(i)}, l(\mathbf{x}^{(i)}))\}_{i=1}^N$ using a noiseless Gaussian Process with the modified kernel $\tilde{k}(\cdot, \cdot)$ yields a vector $\tilde{\lambda}$ such that the function $\tilde{\mu}(\cdot) = \tilde{\mathbf{K}}^T(\cdot) \tilde{\lambda}$ equals the immediate cost function $l(\mathbf{x})$ at the training samples $\mathbf{x}^{(i)}$, $i = 1, \dots, N$. Due to the definition of $\tilde{k}(\cdot, \cdot)$ through the modified feature map $\boldsymbol{\psi}(\cdot)$, it follows that

$$\tilde{\mathbf{K}}(\cdot) = \begin{bmatrix} \langle \boldsymbol{\psi}(\mathbf{x}^{(1)}), \boldsymbol{\psi}(\cdot) \rangle \\ \vdots \\ \langle \boldsymbol{\psi}(\mathbf{x}^{(N)}), \boldsymbol{\psi}(\cdot) \rangle \end{bmatrix}. \quad (22)$$

Therefore, we obtain

$$\boldsymbol{\theta} = \sum_{i=1}^N \tilde{\lambda}_i \boldsymbol{\psi}(\mathbf{x}^{(i)}) \quad (23)$$

due to linearity of the scalar product. Finally, the approximation of the infinite horizon cost function is obtained by substituting (23) in (16), which results in

$$\tilde{V}(\cdot) = \sum_{i=1}^N \tilde{\lambda}_i \left(k(\mathbf{x}^{(i)}, \cdot) - \gamma k(\mathbf{f}(\mathbf{x}^{(i)}), \cdot) \right) \quad (24)$$

due to linearity of the scalar product. ■

Note that even though we used Mercer's theorem to express the kernel $k(\cdot, \cdot)$ in terms of a feature map $\phi(\cdot)$, this decomposition is only necessary for the proof of Theorem 2. It is not necessary to explicitly calculate or evaluate the feature map $\phi(\cdot)$, and only kernel evaluations are required for learning the infinite horizon cost function $\tilde{V}(\cdot)$.

Furthermore, note that the decrease condition

$$\Delta\tilde{V}(\mathbf{x}) = \tilde{V}(\mathbf{x}) - \tilde{V}(\mathbf{f}(\mathbf{x})) > \epsilon \quad (25)$$

in (5) can also be expressed as a linear combination of kernel terms. This is achieved by substituting (24) in (25) to obtain

$$\Delta\tilde{V}(\mathbf{x}) = \tilde{\lambda}^T \hat{\mathbf{k}}(\mathbf{x}) \quad (26)$$

with the linear combination of kernel expressions

$$\begin{aligned} \hat{\mathbf{k}}(\mathbf{x}, \mathbf{x}') &= k(\mathbf{x}, \mathbf{x}') - \gamma k(\mathbf{f}(\mathbf{x}), \mathbf{x}') - k(\mathbf{x}, \mathbf{f}(\mathbf{x}')) \\ &\quad + \gamma k(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{x}')). \end{aligned} \quad (27)$$

Finally, note that Theorem 2 allows to regress both undiscounted and discounted cost functions. However, a cost function with $\gamma < 1$ exhibits two main advantages over the undiscounted case. Infinite horizon cost functions with $\gamma = 1$ are infinite for unstable trajectories and increasing immediate costs $l(\cdot)$, while the discounted infinite horizon cost can be well defined for such systems. Furthermore, the learning method proposed in Theorem 2 can only enforce the Bellman equation locally. While this implies that the learned approximation $\tilde{V}(\mathbf{x})$ is well defined even if $V(\mathbf{x})$ is infinite, it causes a decreasing infinite horizon cost along diverging trajectories. This leads to small values of c in (6), and thus small ROAs \mathbb{V} . However, a suitable discount factor can avoid this behavior. The following Lemma formalizes these advantages and provides sufficient conditions.

Lemma 2: Consider a dynamical system $\mathbf{f}(\cdot)$ and an immediate cost function $l(\cdot)$, such that

$$l(\mathbf{f}(\mathbf{x})) \leq a l(\mathbf{x}) \quad \forall \mathbf{x} \in \mathbb{R}^d \quad (28)$$

for some $a \in \mathbb{R}_+$. Then, the discounted infinite horizon cost function $V(\cdot)$ is finite on \mathbb{R}^d if $\gamma < \frac{1}{a}$. Furthermore, $V(\cdot)$ does not decrease along monotonically diverging trajectories, i.e., trajectories satisfying $l(\mathbf{x}_{k+1}) \geq l(\mathbf{x}_k)$, $\forall k \in \mathbb{N}$.

Proof: Due to the bounded divergence of the system trajectories, the discounted infinite horizon cost is upper bounded through the geometric series with common ratio $a\gamma < 1$ by assumption. Therefore, we obtain

$$V(\mathbf{x}) \leq \frac{l(\mathbf{x})}{1 - a\gamma} \quad (29)$$

which shows that $V(\cdot)$ is properly defined for all $\mathbf{x} \in \mathbb{R}^d$. Furthermore, it is trivial to show that the cost function does not decrease along a trajectory starting at a point \mathbf{x} if

$$l(\mathbf{x}) \leq (1 - \gamma)V(\mathbf{f}(\mathbf{x})). \quad (30)$$

If \mathbf{x} is a point of a monotonically diverging trajectory, we can lower bound the infinite horizon cost by

$$V(\mathbf{f}(\mathbf{x})) \geq \frac{l(\mathbf{x})}{1 - \gamma}, \quad (31)$$

Algorithm 1: Decrease Region Computation

```

1 Function DR( $\Delta V(\cdot)$ ,  $L_{\Delta V}(\cdot, \cdot)$ ,  $\epsilon$ ,  $\mathbb{X}$ ,  $N$ ,  $\mathbf{f}(\cdot)$ ,  $\eta_1$ ):
2   [ $\mathbf{X}$ ,  $\mathbf{B}$ ]  $\leftarrow$  ReachGrid( $\mathbb{X}$ ,  $N$ ,  $\mathbf{f}(\cdot)$ );
3    $\mathbf{d} \leftarrow$  False(length( $\mathbf{X}$ ));  $\mathbf{s} \leftarrow$  True(length( $\mathbf{X}$ ));
4   while any( $s_i$ ) = True do
5     [ $\mathbf{X}$ ,  $\mathbf{B}$ ]  $\leftarrow$  Resample( $\mathbf{X}$ ,  $\mathbf{B}$ ,  $\mathbf{s}$ );
6     for  $i = 1 : \text{length}(\mathbf{X})$  do
7        $v \leftarrow \Delta V(\mathbf{x}_i)$ ;  $v^\partial \leftarrow L_{\Delta V}(\mathbf{x}_i, \mathbf{b}_i)$ ;
8        $d_i \leftarrow (v - \|\mathbf{b}_i\|v^\partial \geq \epsilon) \wedge (\mathbf{x}_i \in \mathbb{X})$ ;
9        $s_i \leftarrow \neg (d_i \vee (\|\mathbf{b}_i\|v^\partial \leq \eta_1)) \vee (\mathbf{x}_i \notin \mathbb{X})$ ;
10  return  $\mathbf{X}$ ,  $\mathbf{B}$ ,  $\mathbf{d}$ 

```

hence the infinite horizon cost does not decrease. ■

Although it might seem that (28) is a restrictive condition on the allowed system dynamics $\mathbf{f}(\cdot)$, it is always possible to find a constant a for continuous dynamics such that (28) holds on a compact set. Since Theorem 1 aims to extend a known asymptotically stable set \mathbb{S} , the conditions of Theorem 1 only need to be analyzed locally. Hence, only the dynamics on a compact set are relevant and, therefore, the dynamics can be defined everywhere else such that they also satisfy (28) with constant a .

Remark 1: Although Theorem 2 allows to learn an approximate infinite horizon cost function with noiseless GP regression, it is often advantageous to use noisy GP regression with a small noise variance σ_n^2 . This is due to the fact that small noise variances σ_n^2 have a beneficial effect on the numerical stability of the matrix inversion required to compute $\tilde{\lambda}$. Even though the Bellman equation cannot be guaranteed anymore at the sampling points, the error due to the noise variance σ_n^2 is small. This follows from the fact that the mean squared regression error at the training samples is smaller than the noise variance σ_n^2 [13].

Remark 2: In our approach, we do not use the GP posterior variance, which corresponds to the expected mean squared error under the prior GP distribution [13]. However, the posterior variance can be employed in efficient training data selection methods, such as [18].

C. Sampling-Based Analysis of Learned Cost Functions

Due to the complexity of the learned infinite horizon cost $\tilde{V}(\cdot)$ and the difference function $\Delta\tilde{V}(\cdot)$, it is not possible to calculate level sets or decreasing regions trivially. For this reason, we propose to use a sampling based approach similar to [9]. The conceptual idea of this method is to compute upper and lower bounds for $\Delta\tilde{V}(\cdot)$ and $\tilde{V}(\cdot)$ on small intervals inside the analyzed set \mathbb{X} with the help of local Lipschitz constants. Since each interval can be analyzed independently with interval analysis, this approach allows complete parallelization and can be implemented efficiently by iteratively reducing the size of the intervals.

Pseudocode for determining the decrease region \mathbb{D}_ϵ is depicted in Alg. 1. In line 2, the union of the reachable set $\mathbb{Q} = \{\mathbf{x} \in \mathbb{R}^d : \exists \mathbf{x}' \in \mathbb{X} : \mathbf{f}(\mathbf{x}') = \mathbf{x}\}$ and the analyzed region \mathbb{X} is covered by N hyperrectangles with centers concatenated in \mathbf{X} and edge lengths concatenated in \mathbf{B} . In line 5, the sampling is refined by dividing the hyperrectangles

Algorithm 2: Region of Attraction Computation

```

1 Function ROA( $V(\cdot)$ ,  $L_V(\cdot, \cdot)$ ,  $\mathbf{X}$ ,  $\mathbf{B}$ ,  $\mathbf{d}$ ,  $\eta_2$ ):
2    $\mathbf{Z} \leftarrow []$ ;  $\mathbf{C} \leftarrow []$ ;
3   for  $i = 1 : \text{length}(\mathbf{X})$  do
4     if  $d_i = \text{False}$  then
5        $\tilde{\mathbf{X}} \leftarrow [\tilde{\mathbf{X}} \ x_i]$ ;  $\tilde{\mathbf{B}} \leftarrow [\tilde{\mathbf{B}} \ b_i]$ ;
6        $\mathbf{v} \leftarrow [\mathbf{v} \ V(\mathbf{x}_i)]$ ;
7        $\mathbf{s} = \text{True}(\text{length}(\mathbf{v}))$ ;  $v_{\min} \leftarrow \min_i v_i$ ;
8       while  $\text{any}(\mathbf{s}_i) = \text{True}$  do
9          $[\tilde{\mathbf{X}}, \tilde{\mathbf{B}}] \leftarrow \text{Resample}(\tilde{\mathbf{X}}, \tilde{\mathbf{B}}, \mathbf{s})$ ;
10        for  $i = 1 : \text{length}(\mathbf{Z})$  do
11           $v_i \leftarrow V(\mathbf{z}_i)$ ;  $v^\partial \leftarrow L_V(\mathbf{z}_i, \mathbf{c}_i)$ ;
12           $s_i \leftarrow v_i - \|\mathbf{c}_i\|v^\partial \leq v_{\min} - \eta_2|v_{\min}|$ ;
13         $v_{\min} \leftarrow \min_i v_i$ ;
14 return  $v_{\min}$ 

```

with an indication value $s_i = \text{True}$. Next, the value v at center \mathbf{x}_i and the local Lipschitz constant v^∂ are computed for all hyperrectangles. If the worst case value in the hyperrectangle satisfies the decrease condition (25), it is added to the decrease region \mathbb{D}_ϵ by setting the decrease flag d_i to True in line 8. If the decrease condition is satisfied or the error tolerance η_1 has been reached, the indication value s_i is set to False in line 9, such that this hyperrectangle is not refined in following iterations. This refinement stops when no sample exhibits an indication value $s_i = \text{True}$. Note that states outside the analyzed set \mathbb{X} are always considered unstable, and thus hyperrectangles in this region are not refined.

After the hyperrectangles within the asymptotically stable set \mathbb{S} have been added to the decrease region \mathbb{D} , the level of the ROA can be computed with Alg. 2. For this reason, the hyperrectangles with $d_i = \text{False}$ are extracted in lines 2 - 6. In lines 8 - 13, the grid is refined until the specified relative precision η_2 is reached for each hyperrectangle. The minimum of the values v_i at centers \mathbf{x}_i is returned, which defines the level of the ROA through $c = v_{\min} - \eta_2|v_{\min}|$.

The functions $L_{\Delta V}(\cdot, \cdot)$ and $L_V(\cdot, \cdot)$ used to determine local Lipschitz constants v^∂ are defined with the help of interval analysis [10], which must be adapted in order to suit the special structure of Gaussian processes. For this reason, we derive the following theorem for GPs with a 1-dimensional squared exponential kernel, whose extension to multidimensional state spaces is straightforward.

Theorem 3: Consider the mean $\mu(\cdot) = \mathbf{k}^T(\cdot)\boldsymbol{\lambda}$ of a GP with squared exponential kernel $k(\cdot, \cdot)$ on an interval with center x and length $2b$. Let l denote the length scale of the kernel. Then, a local Lipschitz constant on the interval is given by

$$L_\mu(x, b) = \max \left\{ \left| \sum_{i=1}^N \mathbf{R}(\lambda_i) \lambda_i \mathbf{L}_{k,i} \right| \right\}, \quad (32)$$

where the absolute value is taken element-wise, and

$$\mathbf{R}(\lambda_i) = \begin{cases} \mathbf{I}_2 & \lambda_i > 0 \\ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & \lambda_i \leq 0 \end{cases} \quad (33)$$

$$\mathbf{L}_{k,i} = \begin{cases} \begin{bmatrix} \frac{\partial}{\partial x} k(x^{(i)}, x+b) \\ \frac{\partial}{\partial x} k(x^{(i)}, x-b) \end{bmatrix} & |x - x^{(i)}| > b+l \\ \begin{bmatrix} \frac{\partial}{\partial x} k(x^{(i)}, x-b) \\ \frac{\partial}{\partial x} k(x^{(i)}, x+b) \end{bmatrix} & |x - x^{(i)}| < l-b \\ \begin{bmatrix} \frac{\partial}{\partial x} k(x^{(i)}, x^{(i)}-l) \\ \min\{\frac{\partial}{\partial x} k(x^{(i)}, x \pm b)\} \end{bmatrix} & l-b \leq x^{(i)} - x \leq l+b \\ \begin{bmatrix} \max\{\frac{\partial}{\partial x} k(x^{(i)}, x \pm b)\} \\ \frac{\partial}{\partial x} k(x^{(i)}, x^{(i)}+l) \end{bmatrix} & l-b \leq x - x^{(i)} \leq l+b. \end{cases} \quad (34)$$

Proof: Since the derivative of the squared exponential kernel is monotonous in the intervals $[-\infty, x^{(i)} - l]$, $[x^{(i)} - l, x^{(i)} + l]$ and $[x^{(i)} + l, \infty]$, the first element in $\mathbf{L}_{k,i}$ is an upper bound for the derivative of the kernel inside the interval with center x , whereas the second element is a lower bound. Multiplication with the matrix $\mathbf{R}(\lambda_i)$ changes the order of the elements in $\mathbf{L}_{k,i}$ if the weight λ_i is negative. Hence, upper bounds for the derivatives are multiplied with positive λ_i s and lower bounds are multiplied with negative λ_i s in the first row of this sum. In the second row the inverse combination is summed up. Therefore, the first row contains an upper bound for the derivative of $\mu(\cdot)$ in the considered interval, whereas the second line is a lower bound. By taking the maximum of the absolute values of both rows, a local Lipschitz constant on this interval is obtained. ■

Although the functions $\Delta \tilde{V}(\cdot)$ and $\tilde{V}(\cdot)$ are no GP means, they exhibit the same structure. Therefore, an extension of Theorem 3 to these functions is straightforward, allowing efficient computation of local Lipschitz constants v^∂ .

IV. NUMERICAL EVALUATION

We compare our approach with the method proposed in [9]. We analyze the functions with the same sampling-based method to allow a fair comparison of their flexibility. The performance is evaluated for the system

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} 1.1x_1 \exp\left(-\frac{(x_3-1)^2}{5}\right) \\ 0.8x_2 + 0.5x_3 \sin(\pi x_3) \\ x_3 - 0.1 \tanh(x_3) \tanh(4 - x_1^2 - x_2^2) \end{bmatrix}, \quad (35)$$

which is unstable for some initial states \mathbf{x} and can generate unbounded trajectories. We choose $\mathbb{X} = [-1.5, 1.5]^3$ as analyzed region, which leads to the set $\mathbb{X} \cup \mathbb{Q} = [-1.65, 1.65] \times [-1.5, 1.5] \times [-1.55, 1.55]$. For our approach, a quadratic immediate cost function is evaluated at a uniformly sampled grid over $\mathbb{X} \cup \mathbb{Q}$. A noisy Gaussian process with squared exponential kernel is trained via likelihood maximization to learn the infinite horizon cost with $\gamma = 0.9$, which has been found to provide the largest ROAs experimentally. The method proposed in [9] is executed with the same immediate cost function. Furthermore, the asymptotically stable region $\mathbb{S} = \{\mathbf{x} \in \mathbb{X} : \|\mathbf{x}\| \leq 0.5\}$ is considered in both approaches.

For comparing the computational complexity of both approaches, we execute Alg. 1 and Alg. 2 with $\epsilon = 10^{-10}$, $\eta_1 = 5 \cdot 10^{-2}$, $\eta_2 = 5 \cdot 10^{-2}$ as well as different trajectory

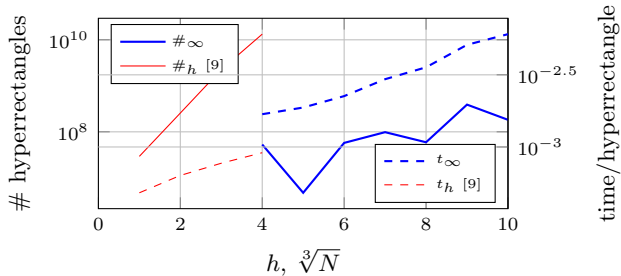


Fig. 1. Comparison of the computational complexity of our method and the approach from [9]

lengths h and number of training samples N . The number of sampled hyperrectangles $\#_h$ for the approach in [9] increases strongly with h . Although the average computation time t_h for the parameters of a single hyperrectangle grows more slowly, tractability limits regarding the overall computation time have been reached with a horizon $h = 4$. In contrast, the average computation time t_∞ is higher for our method, which follows from the fact that the computation of Lipschitz constants requires more mathematical operations. However, the number of analyzed hyperrectangles $\#_\infty$ is far smaller in our method and increases more slowly. Note that we have observed a maximal computation time of 67s for exact inference with GPs, which is negligible compared to the overall runtime of Alg. 1 and Alg. 2. Although the complexity of GP inference scales cubically with the number of training samples N , our approach benefits from the fact that GPs exhibit excellent regression performance with little training data. However, if large training sets are required due to complex dynamical systems, sparse approximations and parallelization can be used to reduce the computational complexity of GP inference [19].

We compare the flexibility of both approaches by fixing $N = 9^3$ training samples in our method and a trajectory length $h = 4$ for the method in [9]. This leads to the ROAs \mathbb{V}_∞ and \mathbb{V}_4 with volumes 2.77 and 0.59, respectively. A cross section through the ROAs along the x_1 - x_3 plane at $x_2 = -0.4$ is depicted in Fig. 2. By simulating 100 steps of our example system on a 1000×1000 grid and analyzing the convergence to the asymptotically stable set \mathbb{S} , we have obtained an estimate of the true ROA, which is also shown in Fig. 2. Although \mathbb{V}_∞ is smaller than the true region of attraction \mathbb{V}_{num} due to regression errors, it is still a large improvement over the set \mathbb{V}_4 obtained with the method in [9].

V. CONCLUSION

This paper presents a novel method for the analysis of local asymptotic stability and the computation of regions of attraction for nonlinear systems. Based on methods from approximate dynamic programming, the infinite horizon cost function is learned with a Gaussian process. By applying a sampling-based interval analysis, the region of attraction can be efficiently computed as a level set of the learned cost.

REFERENCES

[1] A. Chakraborty, P. Seiler, and G. J. Balas, “Nonlinear region of attraction analysis for flight control verification and validation,” *Control Engineering Practice*, vol. 19, no. 4, pp. 335–345, 2011.

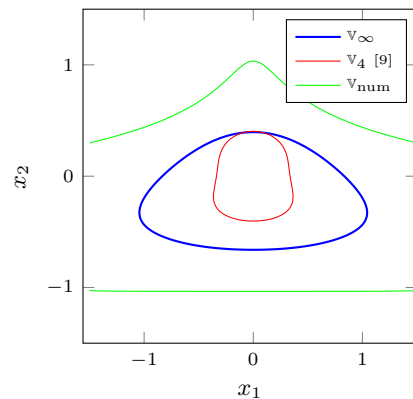


Fig. 2. Cross section along the x_1 - x_3 plane at $x_2 = -0.4$ depicting the ROA estimates

[2] M. Anghel, F. Milano, and A. Papachristodoulou, “Algorithmic construction of Lyapunov functions for power system stability analysis,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 9, pp. 2533–2546, 2013.

[3] A. Merola, C. Cosentino, and F. Amato, “An insight into tumor dormancy equilibrium via the analysis of its domain of attraction,” *Biomedical Signal Processing and Control*, vol. 3, no. 3, pp. 212–219, 2008.

[4] D. Ludwig, B. Walker, and C. S. Holling, “Sustainability, Stability, and Resilience,” *Conservation Ecology*, vol. 1, no. 1, pp. 1–23, 1997.

[5] H. Jerbi, N. B. Braiek, and A. B. B. Bacha, “A Method of Estimating the Domain of Attraction for Nonlinear Discrete-Time Systems,” *Arabian Journal for Science and Engineering*, vol. 39, no. 5, pp. 3841–3849, 2014.

[6] J. Kapinski and J. Deshmukh, “Discovering forward invariant sets for nonlinear dynamical systems,” *Springer Proceedings in Mathematics and Statistics*, vol. 117, pp. 259–264, 2015.

[7] C. K. Luk and G. Chesi, “On the Estimation of the Domain of Attraction for Discrete-Time Switched and Hybrid Nonlinear Systems,” *International Journal of Systems Science*, vol. 46, no. 15, pp. 2781–2787, 2015.

[8] P. Giesl and S. Hafstein, “Computation of Lyapunov functions for nonlinear discrete systems by linear programming,” *Journal of Difference Equations and Applications*, vol. 20, no. 4, pp. 610–640, 2014.

[9] R. Bobiti and M. Lazar, “Automated-Sampling-Based Stability Verification and DOA Estimation for Nonlinear Systems,” *IEEE Transactions on Automatic Control*, vol. 63, no. 11, pp. 3659–3674, 2018.

[10] G. Alefeld and G. Mayer, “Interval analysis: theory and applications,” *Journal of Computational and Applied Mathematics*, vol. 121, pp. 421–464, 2000.

[11] V. Gaitsgory, L. Grüne, M. Höger, C. M. Kellett, and S. R. Weller, “Stabilization of strictly dissipative discrete time systems with discounted optimal control,” *Automatica*, vol. 93, pp. 311–320, 2018.

[12] B. Bethke and J. P. How, “Approximate dynamic programming using Bellman residual elimination and Gaussian process regression,” in *2009 American Control Conference*, 2009, pp. 745–750.

[13] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. The MIT Press, 2006.

[14] I. Steinwart, “On the Influence of the Kernel on the Consistency of Support Vector Machines,” *Journal of Machine Learning Research*, vol. 2, pp. 67–93, 2001.

[15] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2012.

[16] H. K. Khalil, *Nonlinear Systems; 3rd ed.* Upper Saddle River, NJ: Prentice-Hall, 2002.

[17] J. Mercer, “Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 209, no. 441–458, pp. 415–446, 1909.

[18] A. Krause, A. Singh, and C. Guestrin, “Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies,” *Journal of Machine Learning Research*, vol. 9, pp. 235–284, 2008.

[19] R. B. Gramacy, J. Niemi, and R. M. Weiss, “Massively parallel approximate Gaussian process regression,” *SIAM/ASA Journal on Uncertainty Quantification*, vol. 2, no. 1, pp. 564–584, 2014.