Is Data All That Matters? The Role of Control Frequency for Learning-Based Sampled-Data Control of Uncertain Systems

Ralf Römer, Lukas Brunke, Siqi Zhou, and Angela P. Schoellig

***Abstract*— Learning models or control policies from data has become a powerful tool to improve the performance of uncertain systems. While a strong focus has been placed on increasing the amount and quality of data to improve performance, data can never fully eliminate uncertainty, making feedback necessary to ensure stability and performance. We show that the control frequency at which the input is recalculated is a crucial design parameter, yet it has hardly been considered before. We address this gap by combining probabilistic model learning and sampled-data control. We use Gaussian processes (GPs) to learn a continuous-time model and compute a corresponding discrete-time controller. The result is an uncertain sampled-data control system, for which we derive robust stability conditions. We formulate semidefinite programs to compute the minimum control frequency required for stability and to optimize performance. As a result, our approach enables us to study the effect of both control frequency and data on stability and closed-loop performance. We show in numerical simulations of a quadrotor that performance can be improved by increasing either the amount of data or the control frequency, and that we can trade off one for the other. For example, by increasing the control frequency by 33%, we can reduce the number of data points by half while still achieving similar performance.**

I. Introduction

Real-world systems such as robots can exhibit complex dynamics, making deriving accurate models from first principles difficult. Therefore, many studies in recent years have addressed learning unknown dynamics from measured data using machine learning methods and designing a controller based on the learned model [1]–[4]. Much attention has been paid to the role of data and increasing its amount and quality [5]. However, no derived or learned model can perfectly capture a real system's dynamic behavior [6]. Therefore, feedback is required to guarantee stability and performance. The control frequency at which system measurements are fed back to recalculate the control input is often set without taking the dynamics and uncertainty into account [1], neglecting that it represents a degree of freedom in the controller design. However, considering the control frequency as a design parameter can be advantageous, especially for systems such as resource-constrained robot platforms (e.g., drones). For example, knowledge of the minimum control frequency (MCF) required for guaranteed stability of an uncertain system can help improve energy efficiency by reducing unnecessary computational demand and data transmission. In this work, we study the effect of both control frequency and data on closed-loop performance.

The authors are with the Learning Systems and Robotics Lab (learnsyslab.org), School of Computation, Information and Technology, and the Munich Institute for Robotics and Machine Intelligence (MIRMI), Technical University of Munich, Germany. Email: {ralf.roemer; lukas.brunke; siqi.zhou; angela.schoellig}@tum.de.

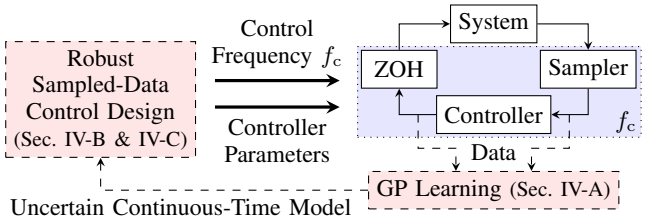

Fig. 1: In digital control systems (blue shaded box), the sampler, controller, and zero-order-hold (ZOH) operate at a certain control frequency f_c. We propose a framework (dashed-line boxes) to simultaneously compute the minimum control frequency and design a controller using an uncertain model learned from data using Gaussian process (GP) regression.

To quantify the uncertainty inherent in a learned dynamics model, probabilistic methods such as Bayesian linear regression (BLR) [7] or Gaussian processes (GPs) [8] have become popular [4]. While BLR assumes linearity in a set of parameters, GPs have the advantage of being a non-parametric method. In [1]–[3], GPs are combined with robust control methods. These works demonstrate that control performance is impacted by model uncertainty and, thus, the training data, but they do not investigate the role of the control frequency.

Few studies have considered the control frequency in the context of model uncertainty. In [9], the maximum sampling interval for stabilizing an unknown linear system is computed directly from data. However, the approach assumes bounded noise and involves a computationally expensive iterative optimization scheme for controller design. GP-based feedback linearization with a data-dependent delay for updating the control input is considered in [10]. It is shown empirically that in terms of tracking accuracy, there can be a tradeoff between the accuracy of the GP model and the computational delay. In [11], reducing the control frequency is discussed in the context of reinforcement learning as a way to reduce the sample complexity and thus improve performance.

Learning a discrete-time model of unknown dynamics yields a model specific to a particular sampling time. Analyzing the system's behavior and stability for a different controller sampling time is generally very difficult. Thus, we consider learning a continuous-time model in this work. Designing a discrete-time controller for a continuous-time system falls within the domain of sampled-data control. The stability of sampled-data systems can be analyzed via the time-delay approach [12], and stability conditions are derived in [13]–[15]. These conditions can also guarantee robust stability for polytopic-type uncertainty, but this becomes

computationally intractable for the uncertainty set associated with a dynamics model learned from data.

We propose a framework to design the control frequency based on the uncertainty associated with a learned dynamics model as illustrated in Fig. 1 and study the role of the control frequency compared to the amount of data. Taking a sampled-data control approach, we robustly stabilize a partially unknown nonlinear continuous-time system with a discrete-time controller. Our main contributions are:

- We combine GP-based stochastic model learning with sampled-data control to study the effect of the control frequency and model uncertainty on the closed-loop performance.
- We derive robust stability conditions as matrix inequalities for a sampled-data control system with learned uncertain dynamics. Based on these, we formulate semidefinite programs (SDPs) for the computation of the MCF and performance optimization. Our framework enables us to control the system at different frequencies without having to re-learn the model.
- Through numerical simulations[1], we show and analyze the tradeoff between model uncertainty, affected by the amount of data collected, and control frequency in terms of stability and performance.

Our results demonstrate that the choice of the control frequency can be as crucial as collecting more data (to further reduce uncertainty).

Notation: We denote the Kronecker product by $\otimes$, the Hadamard (element-wise) product by $\circ$ and the probability by $\Pr(\cdot)$. In symmetric matrices, $*$ denotes transpose elements that can be inferred from symmetry. Given a square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\mathrm{diag}(\boldsymbol{A}) \in \mathbb{R}^n$ is a vector containing the diagonal elements of $\boldsymbol{A}$. Given a vector $\boldsymbol{a} \in \mathbb{R}^n$, $\mathrm{Diag}(\boldsymbol{a}) \in \mathbb{R}^{n\times n}$ is a diagonal matrix containing the elements of $\boldsymbol{a}$ on its diagonal. Given a matrix $\boldsymbol{B} = \begin{bmatrix}\boldsymbol{b}_1 & \dots & \boldsymbol{b}_m\end{bmatrix} \in \mathbb{R}^{n\times m}$, we denote its vectorization by $\mathrm{vec}(\boldsymbol{B}) = \begin{bmatrix}\boldsymbol{b}_1^\mathsf{T} & \dots & \boldsymbol{b}_m^\mathsf{T}\end{bmatrix}^\mathsf{T} \in \mathbb{R}^{nm}$.

II. Problem Statement

We consider a dynamical system whose state and input at time $t \in \mathbb{R}_{\geq 0}$ are given by $\boldsymbol{x}(t) \in \mathbb{R}^n$ and $\boldsymbol{u}(t) \in \mathbb{R}^m$, respectively. The system evolves according to the dynamics

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t)) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) + \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \quad (1)$$

where the function $\boldsymbol{f} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is known, for example, derived from first principles, and the function $\boldsymbol{g} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is unknown, accounting for unmodeled dynamic effects. Both $\boldsymbol{f}$ and $\boldsymbol{g}$ are assumed to be continuously differentiable. We denote $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{u}) \in \mathbb{R}^{n_z}$, where $n_z = n+m$, for brevity. We assume the availability of noisy measurement data collected from system (1).

Assumption 1: A dataset of N observations from (1)

$$\mathcal{D} = \left\{\boldsymbol{z}^{(i)}, \boldsymbol{y}^{(i)} = \dot{\boldsymbol{x}}^{(i)} - \boldsymbol{f}\left(\boldsymbol{z}^{(i)}\right) + \boldsymbol{w}^{(i)}\right\}_{i=1}^{N} \quad (2)$$

is available, with targets perturbed by independent and identically distributed (i.i.d.) Gaussian noise $\boldsymbol{w}^{(i)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_\mathrm{n})$.

While this assumption requires exact state and input measurements, it allows for Gaussian perturbed observations of the state derivative, which is often approximated via finite differences in practice. Similar assumptions are made, for example, in [5], [10].

The samples in $\mathcal{D}$ serve as training data to learn an approximation of $\boldsymbol{g}$, denoted by $\hat{\boldsymbol{g}}$. We aim to control system (1) robustly around a known equilibrium $\boldsymbol{z}_\mathrm{e} = (\boldsymbol{x}_\mathrm{e}, \boldsymbol{u}_\mathrm{e})$ with $\boldsymbol{0} = \boldsymbol{f}(\boldsymbol{z}_\mathrm{e}) + \boldsymbol{g}(\boldsymbol{z}_\mathrm{e})$ despite the uncertainty associated to $\hat{\boldsymbol{g}}$. This does not represent a significant restriction as we can estimate any unknown equilibrium $\boldsymbol{z}_\mathrm{e}$ by solving a nonlinear optimization problem involving $\boldsymbol{f}$ and $\hat{\boldsymbol{g}}$, cf. [1]. We use a discrete-time controller with zero-order-hold

$$\boldsymbol{u}(t) = \boldsymbol{\pi}(\boldsymbol{x}(t_k)), \qquad \forall t \in [t_k, t_{k+1}), \quad (3)$$

where $\boldsymbol{\pi}(\cdot)$ is a control law based on the learned continuous-time model, and t_k, $k \in \mathbb{N}_0$, $t_0 = 0$, are the sampling instants. We make an assumption on the sampling instants capturing periodic and aperiodic sampling and define the MCF.

Assumption 2: The interval between two consecutive sampling instants satisfies $t_{k+1} - t_k \leq T_\mathrm{s}$, $\forall k \in \mathbb{N}_0$, where T_s is an upper bound on the sampling interval.

Definition 1: Let $T_{\mathrm{s,max}}$ be the largest value of T_s such that the system (1) with the control (3) satisfying Assumption 2 can be robustly stabilized around $\boldsymbol{z}_\mathrm{e}$. Then, the MCF is given by $f_{\mathrm{c,min}} = \frac{1}{T_{\mathrm{s,max}}}$.

The MCF depends on the employed feedback control law $\boldsymbol{\pi}(\cdot)$. We consider the problem of learning a local linear approximation of $\boldsymbol{g}$ from the data (2), computing the MCF together with a linear control law based on the uncertainty of the learned dynamics and aim to optimize the control performance. Ultimately, we want to study the impact of the control frequency on stability and performance, especially compared to model uncertainty and the amount of data.

III. Background

A. Gaussian Process Regression

To simplify notation, we consider learning an unknown scalar function $g : \mathbb{R}^{n_z} \to \mathbb{R}$ from training inputs $\boldsymbol{z}^{(i)}$ and target $y^{(i)} = g(\boldsymbol{z}^{(i)}) + w^{(i)}$, $i = 1, \dots, N$, which are perturbed by i.i.d. noise $w^{(i)} \sim \mathcal{N}(0, \sigma_\mathrm{n}^2)$. GP regression [8] assumes that the unknown function is drawn from a GP, denoted as $\mathcal{GP}(\mu(\cdot), k(\cdot,\cdot))$, which induces a distribution over functions such that any finite number of function evaluations is jointly Gaussian distributed. The prior mean function $\mu : \mathbb{R}^{n_z} \to \mathbb{R}$ can incorporate prior knowledge in the form of an approximate model, and the kernel $k : \mathbb{R}^{n_z} \times \mathbb{R}^{n_z} \to \mathbb{R}$ encodes information about the structure of the unknown function. Without loss of generality, we set the mean function to zero. Under the GP assumption, the vector of observed targets $\boldsymbol{y} = \begin{bmatrix}y^{(1)}, \dots, y^{(N)}\end{bmatrix}^\mathsf{T}$ and the function value at a query point $\boldsymbol{z}^* \in \mathbb{R}^{n_z}$ have the joint probability distribution

$$\begin{bmatrix}\boldsymbol{y} \\ g(\boldsymbol{z}^*)\end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix}\bar{\boldsymbol{K}} & \boldsymbol{k}(\boldsymbol{z}^*)^\mathsf{T} \\ * & k(\boldsymbol{z}^*, \boldsymbol{z}^*)\end{bmatrix}\right), \quad (4)$$

[1]All code for reproducing the results reported in this paper is available at https://github.com/ralfroemer99/lb_sd

where the gram matrix $\bar{\boldsymbol{K}}$ and vector $\boldsymbol{k}(\boldsymbol{z}^*)$ are defined as $\bar{\boldsymbol{K}} = \boldsymbol{K} + \sigma_{\mathrm{n}}^2 \boldsymbol{I}_n$, where $\boldsymbol{K}_{ij} = k\left(\boldsymbol{z}^{(i)}, \boldsymbol{z}^{(j)}\right)$, and $\boldsymbol{k}(\boldsymbol{z}^*) = \left[k\left(\boldsymbol{z}^{(1)}, \boldsymbol{z}^*\right), \ldots, k\left(\boldsymbol{z}^{(N)}, \boldsymbol{z}^*\right)\right]^{\mathsf{T}}$, respectively. Conditioning $g(\boldsymbol{z}^*)$ on the training data yields the posterior predictive distribution $g(\boldsymbol{z}^*) \sim \mathcal{N}(\mu(\boldsymbol{z}^*), \sigma^2(\boldsymbol{z}^*))$ with mean $\mu(\boldsymbol{z}^*) = \boldsymbol{k}(\boldsymbol{z}^*)^{\mathsf{T}} \bar{\boldsymbol{K}}^{-1} \boldsymbol{y}$ and variance $\sigma^2(\boldsymbol{z}^*) = k(\boldsymbol{z}^*, \boldsymbol{z}^*) - \boldsymbol{k}(\boldsymbol{z}^*)^{\mathsf{T}} \bar{\boldsymbol{K}}^{-1} \boldsymbol{k}(\boldsymbol{z}^*)$.

As the derivative is a linear operator, the derivative of a GP is also a GP [8]. We can use this property to predict the derivative of g at $\boldsymbol{z}^*$, denoted $\left.\frac{\partial g(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*}$. From (4), we obtain

$$\begin{bmatrix} \boldsymbol{y} \\ \left.\frac{\partial g(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*} \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} \bar{\boldsymbol{K}} & \left.\frac{\partial \boldsymbol{k}(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*} \\ * & \left.\frac{\partial^2 k(\boldsymbol{z},\boldsymbol{z})}{\partial \boldsymbol{z} \partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*} \end{bmatrix}\right). \tag{5}$$

Similar to the derivation of $g(\boldsymbol{z}^*)$ from (4), conditioning $\left.\frac{\partial g(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*}$ on the observations $\boldsymbol{y}$ yields the predictive distribution $\left.\frac{\partial g(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*} \sim \mathcal{N}(\boldsymbol{\mu}'(\boldsymbol{z}^*), \boldsymbol{\Sigma}'(\boldsymbol{z}^*))$ with mean and variance

$$\boldsymbol{\mu}'(\boldsymbol{z}^*) = \left.\frac{\partial \boldsymbol{k}(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*} \bar{\boldsymbol{K}}^{-1} \boldsymbol{y}, \tag{6}$$

$$\boldsymbol{\Sigma}'(\boldsymbol{z}^*) = \left.\frac{\partial^2 k(\boldsymbol{z},\boldsymbol{z})}{\partial \boldsymbol{z} \partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*} - \left.\frac{\partial \boldsymbol{k}(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*} \bar{\boldsymbol{K}}^{-1} \left(\left.\frac{\partial \boldsymbol{k}(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^*}\right)^{\mathsf{T}}. \tag{7}$$

B. Sampled-Data Systems

Consider a continuous-time LTI system

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{8}$$

under a discrete-time linear state-feedback control law

$$\boldsymbol{u}(t) = \boldsymbol{K}\boldsymbol{x}(t_k), \qquad \forall t \in [t_k, t_{k+1}), \tag{9}$$

where $\boldsymbol{K} \in \mathbb{R}^{m \times n}$, and the sampling instants t_k satisfy Assumption 2. The time-delay approach [12]–[15] to sampled-data systems writes the closed-loop system as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{K}\boldsymbol{x}(t - \tau(t)), \tag{10}$$

where $\tau(t) = t - t_k$, $\forall t \in [t_k, t_{k+1})$, is a piecewise-continuous, time-varying delay. Stability of (10) can be analyzed using the Lyapunov-Krasovskii functional [13]

$$V(t) = \boldsymbol{x}(t)^{\mathsf{T}} \boldsymbol{P}_1 \boldsymbol{x}(t) + \int_{-T_{\mathrm{s}}}^{0} \int_{t+\theta}^{t} \dot{\boldsymbol{x}}(s)^{\mathsf{T}} \boldsymbol{P}_2 \dot{\boldsymbol{x}}(s) \mathrm{d}s \mathrm{d}\theta, \tag{11}$$

where $\boldsymbol{P}_1 \succ \boldsymbol{0}$, $\boldsymbol{P}_2 \succ \boldsymbol{0}$. The second term in (11) is used to compensate for the delay-dependent part in $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}(t)^{\mathsf{T}} \boldsymbol{P}_1 \boldsymbol{x}(t)$.

Lemma 1 ([13], Lemma 2.3): The control (9) asymptotically stabilizes system (8) for all samplings satisfying Assumption 2 if there exist matrices $\boldsymbol{Q}_1 = \boldsymbol{Q}_1^{\mathsf{T}} \succ \boldsymbol{0}$, $\boldsymbol{Q}_2$, $\boldsymbol{Q}_3$, $\boldsymbol{Z}_1$, $\boldsymbol{Z}_2$, $\boldsymbol{Z}_3$, $\boldsymbol{R} = \boldsymbol{R}^{\mathsf{T}} \succ \boldsymbol{0}$, all in $\mathbb{R}^{n \times n}$, and $\boldsymbol{Y} \in \mathbb{R}^{m \times n}$, that satisfy the matrix inequalities

$$\boldsymbol{W}_{\boldsymbol{A},\boldsymbol{B}} \triangleq \begin{bmatrix} \boldsymbol{\Xi} & \boldsymbol{\Xi}_{\boldsymbol{A},\boldsymbol{B}} & T_{\mathrm{s}} \boldsymbol{Q}_2^{\mathsf{T}} \\ * & -\boldsymbol{Q}_3 - \boldsymbol{Q}_3^{\mathsf{T}} + T_{\mathrm{s}} \boldsymbol{Z}_3 & T_{\mathrm{s}} \boldsymbol{Q}_3^{\mathsf{T}} \\ * & * & -T_{\mathrm{s}} \boldsymbol{R} \end{bmatrix} \prec \boldsymbol{0}, \tag{12}$$

$$\begin{bmatrix} \boldsymbol{Q}_1 \boldsymbol{R}^{-1} \boldsymbol{Q}_1 & \boldsymbol{0} & \boldsymbol{Y}^{\mathsf{T}} \boldsymbol{B}^{\mathsf{T}} \\ * & \boldsymbol{Z}_1 & \boldsymbol{Z}_2 \\ * & * & \boldsymbol{Z}_3 \end{bmatrix} \succeq \boldsymbol{0}, \tag{13}$$

where $*$ denotes symmetry, $\boldsymbol{\Xi} = \boldsymbol{Q}_2 + \boldsymbol{Q}_2^{\mathsf{T}} + T_{\mathrm{s}} \boldsymbol{Z}_1$ and $\boldsymbol{\Xi}_{\boldsymbol{A},\boldsymbol{B}} = \boldsymbol{Q}_3 - \boldsymbol{Q}_2^{\mathsf{T}} + \boldsymbol{Q}_1 \boldsymbol{A}^{\mathsf{T}} + T_{\mathrm{s}} \boldsymbol{Z}_2 + \boldsymbol{Y}^{\mathsf{T}} \boldsymbol{B}^{\mathsf{T}}$. The stabilizing state-feedback gain is then given by $\boldsymbol{K} = \boldsymbol{Y}\boldsymbol{Q}_1^{-1}$.

Proof: The proof involves multiple steps, including defining a descriptor system of (10), differentiating (11) for time and using that the delay is bounded by $\tau(t) \in [0, T_{\mathrm{s}}]$ due to Assumption 2. See [12], [13] for details. ∎

IV. Methodology

In this section, we first discuss learning a probabilistic estimate of the dynamics (1) and a locally valid uncertain linearization from the dataset (2). Then, we present our approach for robust sampled-data control for different control frequencies based on the learned model's uncertainty.

A. Model Learning and Linearization

We train a GP model for each output dimension of $\boldsymbol{g}(\cdot) = [g_1(\cdot), \ldots, g_n(\cdot)]^{\mathsf{T}}$, assuming the following:

Assumption 3: The functions g_i, $i = 1, \ldots, n$, are drawn from zero-mean GPs with squared-exponential (SE) kernel

$$k_i(\boldsymbol{z}, \boldsymbol{z}') = \sigma_{\eta,i}^2 \exp\left(-\frac{1}{2}(\boldsymbol{z} - \boldsymbol{z}')^{\mathsf{T}} \boldsymbol{L}_i^{-2} (\boldsymbol{z} - \boldsymbol{z}')\right), \tag{14}$$

where $\sigma_{\eta,i}^2 > 0$ and $\boldsymbol{L}_i = \mathrm{Diag}(\boldsymbol{l}_i) \in \mathbb{R}^{n \times n}$, $\boldsymbol{l}_i > \boldsymbol{0} \in \mathbb{R}^n$.

In (14), $\sigma_{\eta,i}^2$ is the output variance, and $\boldsymbol{L}_i$ contains the vector of length scales $\boldsymbol{l}_i$, which corresponds to the rate of change of g_i with respect to $\boldsymbol{z}$. Assumption 3 is not restrictive in practice as the corresponding space of sample functions of the GP contains all continuous functions [16]. The kernel hyperparameters are often unknown, but they can be determined, for example, by maximizing the marginal log-likelihood of the training data (2); see [8]. We compute the derivative of the i-th GP at $\boldsymbol{z}_{\mathrm{e}}$, denoted by $\left.\frac{\partial g_i(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}_{\mathrm{e}}} \sim \mathcal{N}\left(\boldsymbol{\mu}_i'(\boldsymbol{z}_{\mathrm{e}}), \boldsymbol{\Sigma}_i'(\boldsymbol{z}_{\mathrm{e}})\right)$, via (6) and (7), where the partial derivatives of (14) are straightforward to evaluate. This gives a probabilistic estimate of the linearized dynamics.

Lemma 2: Under Assumption 3 and given the data (2), the linearization of (1) about $\boldsymbol{z}_{\mathrm{e}}$ satisfies that for any $p \in [0, 1)$,

$$\Pr\left(\left.\frac{\boldsymbol{h}}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}_{\mathrm{e}}} \in \mathcal{C}\right) \geq p^n, \tag{15}$$

where $\mathcal{C} = \left[\hat{\boldsymbol{C}} - \bar{\boldsymbol{C}}, \hat{\boldsymbol{C}} + \bar{\boldsymbol{C}}\right] \subset \mathbb{R}^{n \times n_z}$, where

$$\hat{\boldsymbol{C}} = \begin{bmatrix} \hat{\boldsymbol{A}} & \hat{\boldsymbol{B}} \end{bmatrix} = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}_{\mathrm{e}}} + \begin{bmatrix} \boldsymbol{\mu}_1'(\boldsymbol{z}_{\mathrm{e}})^{\mathsf{T}} \\ \vdots \\ \boldsymbol{\mu}_n'(\boldsymbol{z}_{\mathrm{e}})^{\mathsf{T}} \end{bmatrix}, \tag{16}$$

$$\bar{\boldsymbol{C}} = \begin{bmatrix} \bar{\boldsymbol{A}} & \bar{\boldsymbol{B}} \end{bmatrix} = \sqrt{\chi_{n_z}^2(p)} \begin{bmatrix} \sqrt{\mathrm{diag}(\boldsymbol{\Sigma}_1'(\boldsymbol{z}_{\mathrm{e}}))^{\mathsf{T}}} \\ \vdots \\ \sqrt{\mathrm{diag}(\boldsymbol{\Sigma}_n'(\boldsymbol{z}_{\mathrm{e}}))^{\mathsf{T}}} \end{bmatrix}. \tag{17}$$

Here, $\chi_{n_z}^2$ is the quantile function of the chi-squared distribution of degree n_z.

Proof: The fundamental properties of multivariate Gaussian distributions imply that for all $p \in (0, 1]$, $\Pr\left(\left.\frac{\partial g_i(\boldsymbol{z})}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}_{\mathrm{e}}} - \boldsymbol{\mu}_i'(\boldsymbol{z}_{\mathrm{e}}) \in \mathcal{E}_i \middle| \mathcal{D}\right) = p$, where $\mathcal{E}_i$

is an ellipsoidal confidence region defined by $\mathcal{E}_i = \left\{ \boldsymbol{d} \in \mathbb{R}^{n_z} \middle| \boldsymbol{d}^\mathsf{T} \left(\boldsymbol{\Sigma}_i'(\boldsymbol{z}_\mathrm{e})\right)^{-1} \boldsymbol{d} \leq \chi^2_{n_z}(p) \right\}$, $i = 1, \ldots, n$. The result follows from the independence of the GPs and the fact that $\mathcal{E}_i \subseteq \mathcal{B}_i$, where $\mathcal{B}_i$ is a hyperrectangle with dimensions $2\sqrt{\chi^2_{n_z}(p) \left(\boldsymbol{\Sigma}_i'(\boldsymbol{z}_\mathrm{e})\right)_{jj}}$, $j = 1, \ldots, n_z$, which is symmetric about the origin. ■

Due to Lemma 2, the true linearized dynamics at $\boldsymbol{z} = \boldsymbol{z}_\mathrm{e}$ are captured with probability of at least p^n by

$$\dot{\tilde{\boldsymbol{x}}}(t) = \big(\hat{\boldsymbol{A}} + \bar{\boldsymbol{A}} \circ \boldsymbol{\Omega}\big)\tilde{\boldsymbol{x}}(t) + \big(\hat{\boldsymbol{B}} + \bar{\boldsymbol{B}} \circ \boldsymbol{\Psi}\big)\tilde{\boldsymbol{u}}(t), \tag{18}$$

where $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{x}_\mathrm{e}$, $\tilde{\boldsymbol{u}} = \boldsymbol{u} - \boldsymbol{u}_\mathrm{e}$ are deviations about the equilibrium, and $\boldsymbol{\Omega} \in [-1, 1]^{n \times n}$, $\boldsymbol{\Psi} \in [-1, 1]^{n \times m}$ are unknown.

As $\mathcal{C}$ is a polytopic set, Lemma 1 can be applied to analyze the stability of (18) in principle. However, this requires evaluating the stability conditions (12) and (13) for all 2^{nn_z} vertices of $\mathcal{C}$, which is computationally infeasible except for very low-dimensional systems. The same holds for the stability conditions in [14], [15]. To address this problem, we reparameterize the Hadamard product terms in (18) corresponding to the uncertainty by making use of the following lemma, which is straightforward to prove.

Lemma 3: Let $\boldsymbol{U} = \begin{bmatrix} \boldsymbol{u}_1 & \ldots & \boldsymbol{u}_n \end{bmatrix}^\mathsf{T}, \boldsymbol{V} \in \mathbb{R}^{n \times m}$. Then,

$$\boldsymbol{U} \circ \boldsymbol{V} = (\boldsymbol{I}_n \otimes \boldsymbol{1}_{1\times m}) \mathrm{Diag}\left(\mathrm{vec}\left(\boldsymbol{V}^\mathsf{T}\right)\right) \begin{bmatrix} \mathrm{Diag}(\boldsymbol{u}_1) \\ \vdots \\ \mathrm{Diag}(\boldsymbol{u}_n) \end{bmatrix}.$$

We denote the rows of $\bar{\boldsymbol{A}}$ and $\bar{\boldsymbol{B}}$ by $\bar{\boldsymbol{a}}_1^\mathsf{T}, \ldots, \bar{\boldsymbol{a}}_n^\mathsf{T}$ and $\bar{\boldsymbol{b}}_1^\mathsf{T}, \ldots, \bar{\boldsymbol{b}}_n^\mathsf{T}$, respectively, and define $p = n^2 + nm$. Then, we can use Lemma 3 to rewrite the uncertainty in (18) as

$$\bar{\boldsymbol{A}} \circ \boldsymbol{\Omega} = \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{E}, \qquad \bar{\boldsymbol{B}} \circ \boldsymbol{\Psi} = \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{F}, \tag{19}$$

where $\boldsymbol{\Delta} = \mathrm{Diag}([\delta_1, \ldots, \delta_p]) \in \mathbb{R}^{p \times p}$ with $|\delta_i| \leq 1$, $\forall i$, $\boldsymbol{H} = \big[\boldsymbol{I}_n \otimes \boldsymbol{1}_{1\times n}, \boldsymbol{I}_n \otimes \boldsymbol{1}_{1\times m}\big]$ and

$$\begin{aligned} \boldsymbol{E} &= \begin{bmatrix} \mathrm{Diag}(\bar{\boldsymbol{a}}_1) & \ldots & \mathrm{Diag}(\bar{\boldsymbol{a}}_n) & \boldsymbol{0}_{n\times nm} \end{bmatrix}^\mathsf{T}, \\ \boldsymbol{F} &= \begin{bmatrix} \boldsymbol{0}_{n\times n^2} & \mathrm{Diag}(\bar{\boldsymbol{b}}_1) & \ldots & \mathrm{Diag}(\bar{\boldsymbol{b}}_n) \end{bmatrix}^\mathsf{T}. \end{aligned} \tag{20}$$

Inserting (19) into (18) gives the uncertain linearized system

$$\dot{\tilde{\boldsymbol{x}}}(t) = \big(\hat{\boldsymbol{A}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{E}\big)\tilde{\boldsymbol{x}}(t) + \big(\hat{\boldsymbol{B}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{F}\big)\tilde{\boldsymbol{u}}(t), \tag{21}$$

where all matrices except for $\boldsymbol{\Delta}$ are known.

B. Robust Sampled-Data Control of the Uncertain System

To robustly control the continuous-time system (21), we consider a discrete-time linear state feedback

$$\tilde{\boldsymbol{u}}(t) = \boldsymbol{K}\tilde{\boldsymbol{x}}(t_k), \qquad \forall t \in [t_k, t_{k+1}), \tag{22}$$

where the sampling instants t_k, $k \in \mathbb{N}_0$, satisfy Assumption 2. This results in the uncertain closed-loop system

$$\dot{\tilde{\boldsymbol{x}}}(t) = \big(\hat{\boldsymbol{A}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{E}\big)\tilde{\boldsymbol{x}}(t) + \big(\hat{\boldsymbol{B}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{F}\big)\boldsymbol{K}\tilde{\boldsymbol{x}}(t - \tau(t)), \tag{23}$$

where the delay $\tau(t)$ is defined similar to Section III-B and, thus, bounded by $\tau(t) \in [0, T_\mathrm{s}]$. We employ the following result to take the norm-bounded uncertainty in the linearized GP dynamics model into account:

Lemma 4 ([17]): Let $\boldsymbol{\Theta} \in \mathbb{R}^{m \times m}$ satisfy $\boldsymbol{\Theta}^\mathsf{T}\boldsymbol{\Theta} \preceq \boldsymbol{I}$. Then, for all constant matrices $\boldsymbol{U} \in \mathbb{R}^{n \times m}$, $\boldsymbol{V} \in \mathbb{R}^{m \times n}$ and all scalars $\epsilon > 0$, it holds that

$$\begin{aligned} -\epsilon^{-1}\boldsymbol{U}\boldsymbol{U}^\mathsf{T} - \epsilon\boldsymbol{V}^\mathsf{T}\boldsymbol{V} &\preceq \boldsymbol{U}\boldsymbol{\Theta}\boldsymbol{V} + \boldsymbol{V}^\mathsf{T}\boldsymbol{\Theta}^\mathsf{T}\boldsymbol{U}^\mathsf{T} \\ &\preceq \epsilon^{-1}\boldsymbol{U}\boldsymbol{U}^\mathsf{T} + \epsilon\boldsymbol{V}^\mathsf{T}\boldsymbol{V}. \end{aligned}$$

Using Lemma 4, we derive the following constructive conditions for robust stability of the closed-loop system.

Theorem 1: The uncertain system (23) is robustly asymptotically stable for all samplings satisfying Assumption 2 if there exist matrices $\boldsymbol{Q}_1 = \boldsymbol{Q}_1^\mathsf{T} \succ \boldsymbol{0}$, $\boldsymbol{Q}_2$, $\boldsymbol{Q}_3$, $\boldsymbol{Z}_1$, $\boldsymbol{Z}_2$, $\boldsymbol{Z}_3$, $\boldsymbol{R} = \boldsymbol{R}^\mathsf{T} \succ \boldsymbol{0}$, all in $\mathbb{R}^{n \times n}$, $\boldsymbol{Y} \in \mathbb{R}^{m \times n}$ and scalars $\epsilon_1 > 0$, $\epsilon_2 > 0$, that satisfy the matrix inequalities

$$\left[\begin{array}{ccc|cc} & & & \boldsymbol{0} & \epsilon_1\left(\boldsymbol{Q}_1\boldsymbol{E}^\mathsf{T} + \boldsymbol{Y}^\mathsf{T}\boldsymbol{F}^\mathsf{T}\right) \\ & \boldsymbol{W}_{\hat{\boldsymbol{A}},\hat{\boldsymbol{B}}} & & \boldsymbol{H} & \boldsymbol{0} \\ & & & \boldsymbol{0} & \boldsymbol{0} \\ \hline * & * & * & -\epsilon_1\boldsymbol{I} & \boldsymbol{0} \\ * & * & * & * & -\epsilon_1\boldsymbol{I} \end{array}\right] \prec \boldsymbol{0}, \tag{24}$$

$$\begin{bmatrix} 2\boldsymbol{Q}_1 - \boldsymbol{R} & \boldsymbol{0} & \boldsymbol{Y}^\mathsf{T}\hat{\boldsymbol{B}}^\mathsf{T} & \boldsymbol{0} & \epsilon_2\boldsymbol{Y}^\mathsf{T}\boldsymbol{F}^\mathsf{T} \\ * & \boldsymbol{Z}_1 & \boldsymbol{Z}_2 & \boldsymbol{0} & \boldsymbol{0} \\ * & * & \boldsymbol{Z}_3 & \boldsymbol{H} & \boldsymbol{0} \\ * & * & * & \epsilon_2\boldsymbol{I} & \boldsymbol{0} \\ * & * & * & * & \epsilon_2\boldsymbol{I} \end{bmatrix} \succeq \boldsymbol{0}, \tag{25}$$

where $\boldsymbol{W}_{\hat{\boldsymbol{A}},\hat{\boldsymbol{B}}}$ is defined in (12). The stabilizing state-feedback gain is then given by $\boldsymbol{K} = \boldsymbol{Y}\boldsymbol{Q}_1^{-1}$.

Proof: The idea is to show that if (24) and (25) are satisfied, then the nominal stability conditions (12) and (13) hold for all realizations of the uncertain system matrices $\boldsymbol{A} = \hat{\boldsymbol{A}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{E}$ and $\boldsymbol{B} = \hat{\boldsymbol{B}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{F}$ of (23). We start by applying the Schur complement [18] to the first inequality (24) in Theorem 1, which gives

$$\boldsymbol{W}_{\hat{\boldsymbol{A}},\hat{\boldsymbol{B}}} + \begin{bmatrix} \boldsymbol{M} & \epsilon_1\boldsymbol{N}^\mathsf{T} \end{bmatrix} \begin{bmatrix} \epsilon_1^{-1}\boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \epsilon_1^{-1}\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{M}^\mathsf{T} \\ \epsilon_1\boldsymbol{N} \end{bmatrix} \prec \boldsymbol{0}, \tag{26}$$

where $\boldsymbol{M} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{H}^\mathsf{T} & \boldsymbol{0} \end{bmatrix}^\mathsf{T}$ and $\boldsymbol{N} = \begin{bmatrix} \boldsymbol{E}\boldsymbol{Q}_1^\mathsf{T} + \boldsymbol{F}\boldsymbol{Y} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}$. By rewriting (26) and making use of Lemma 4, we obtain

$$\begin{aligned} &\boldsymbol{W}_{\hat{\boldsymbol{A}},\hat{\boldsymbol{B}}} + \epsilon_1^{-1}\boldsymbol{M}\boldsymbol{M}^\mathsf{T} + \epsilon_1\boldsymbol{N}^\mathsf{T}\boldsymbol{N} \prec \boldsymbol{0} \\ &\boldsymbol{W}_{\hat{\boldsymbol{A}},\hat{\boldsymbol{B}}} + \boldsymbol{M}\boldsymbol{\Delta}\boldsymbol{N} + \boldsymbol{N}^\mathsf{T}\boldsymbol{\Delta}^\mathsf{T}\boldsymbol{M}^\mathsf{T} \prec \boldsymbol{0}, \end{aligned} \tag{27}$$

for all $\boldsymbol{\Delta} \in \mathbb{R}^{p \times p}$ satisfying $\boldsymbol{\Delta}^\mathsf{T}\boldsymbol{\Delta} \preceq \boldsymbol{I}$. Inserting the expressions for $\boldsymbol{M}$, $\boldsymbol{N}$ and $\boldsymbol{W}_{\hat{\boldsymbol{A}},\hat{\boldsymbol{B}}}$ into (27) yields

$$\underbrace{\boldsymbol{W}_{\hat{\boldsymbol{A}},\hat{\boldsymbol{B}}} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{Q}_1(\boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{E})^\mathsf{T} + \boldsymbol{Y}^\mathsf{T}(\boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{F})^\mathsf{T} & \boldsymbol{0} \\ * & \boldsymbol{0} & \boldsymbol{0} \\ * & * & \boldsymbol{0} \end{bmatrix}}_{=\boldsymbol{W}_{\boldsymbol{A},\boldsymbol{B}}} \prec \boldsymbol{0}, \tag{28}$$

which is equivalent to (12). Thus, the satisfaction of (24) implies the satisfaction of (12) for all realizations $\boldsymbol{A} = \hat{\boldsymbol{A}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{E}$, $\boldsymbol{B} = \hat{\boldsymbol{B}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{F}$ of the uncertain system matrices. Consider now the second inequality (25) in Theorem 1. As a result of $\boldsymbol{R} = \boldsymbol{R}^\mathsf{T} \succ \boldsymbol{0}$ and $\boldsymbol{Q}_1 = \boldsymbol{Q}_1^\mathsf{T}$, we have

$$\begin{aligned} (\boldsymbol{Q}_1 - \boldsymbol{R})^\mathsf{T}\boldsymbol{R}^{-1}(\boldsymbol{Q}_1 - \boldsymbol{R}) = \boldsymbol{Q}_1\boldsymbol{R}^{-1}\boldsymbol{Q}_1 - 2\boldsymbol{Q}_1 + \boldsymbol{R} \succ \boldsymbol{0} \\ \Rightarrow \boldsymbol{Q}_1\boldsymbol{R}^{-1}\boldsymbol{Q}_1 \succ 2\boldsymbol{Q}_1 - \boldsymbol{R}. \end{aligned} \tag{29}$$

Hence, the satisfaction of (25) implies

$$\begin{bmatrix} \boldsymbol{Q}_1\boldsymbol{R}^{-1}\boldsymbol{Q}_1 & \boldsymbol{0} & \boldsymbol{Y}^\mathsf{T}\hat{\boldsymbol{B}}^\mathsf{T} & \boldsymbol{0} & \epsilon_2\boldsymbol{Y}^\mathsf{T}\boldsymbol{F}^\mathsf{T} \\ * & \boldsymbol{Z}_1 & \boldsymbol{Z}_2 & \boldsymbol{0} & \boldsymbol{0} \\ * & * & \boldsymbol{Z}_3 & \boldsymbol{H} & \boldsymbol{0} \\ * & * & * & \epsilon_2\boldsymbol{I} & \boldsymbol{0} \\ * & * & * & * & \epsilon_2\boldsymbol{I} \end{bmatrix} \succeq \boldsymbol{0}. \tag{30}$$

Applying the Schur complement to (30) and performing similar steps as above results in

$$\begin{bmatrix} \boldsymbol{Q}_1\boldsymbol{R}^{-1}\boldsymbol{Q}_1 & \boldsymbol{0} & \boldsymbol{Y}^\mathsf{T}(\hat{\boldsymbol{B}}+\boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{F})^\mathsf{T} \\ * & \boldsymbol{Z}_1 & \boldsymbol{Z}_2 \\ * & * & \boldsymbol{Z}_3 \end{bmatrix} \succeq \boldsymbol{0}, \tag{31}$$

for all $\boldsymbol{\Delta}^\mathsf{T}\boldsymbol{\Delta} \preceq \boldsymbol{I}$, which is equivalent to (13) with $\boldsymbol{B} = \hat{\boldsymbol{B}} + \boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{F}$. Thus, if (25) holds, then (13) is satisfied for all realizations of $\boldsymbol{B}$, which concludes the proof. ■

Remark 1: We could also assume $\boldsymbol{R} = \epsilon_3\boldsymbol{Q}_1$ for some $\epsilon_3 > 0$ to convexify the problem [13]. However, this would add another scalar decision variable to Theorem 1, increasing the computational complexity.

Theorem 1 enables us to analyze the stability of the uncertain system (21) for a *given* upper bound on the sampling interval T_s. Moreover, we can compute the MCF $f_{\mathrm{c,min}} = \frac{1}{T_{\mathrm{s,max}}}$ and the corresponding stabilizing control gain $\boldsymbol{K} = \boldsymbol{Y}\boldsymbol{Q}_1^{-1}$ by solving the optimization problem

$$\begin{aligned} \min_{\substack{T_\mathrm{s},\boldsymbol{Q}_1,\boldsymbol{Q}_2,\boldsymbol{Q}_3,\boldsymbol{Z}_1,\\ \boldsymbol{Z}_2,\boldsymbol{Z}_3,\boldsymbol{R},\boldsymbol{Y},\epsilon_1,\epsilon_2}} \quad & \frac{1}{T_\mathrm{s}} \\ \text{s.t.} \quad & (24), \quad (25), \\ & \boldsymbol{Q}_1 = \boldsymbol{Q}_1^\mathsf{T} \succ \boldsymbol{0}, \quad \boldsymbol{R} = \boldsymbol{R}^\mathsf{T} \succ \boldsymbol{0}. \end{aligned} \tag{32}$$

For fixed values of ϵ_1 and ϵ_2, (24) becomes a linear fractional constraint of the form $\lambda\boldsymbol{M}(\boldsymbol{s}) + \boldsymbol{N}(\boldsymbol{s}) \prec \boldsymbol{0}$, where $\lambda = \frac{1}{T_\mathrm{s}} \in \mathbb{R}$ and $\boldsymbol{s} \in \mathbb{R}^{n_s}$ are the optimization variables, and the matrices $\boldsymbol{M}$ and $\boldsymbol{N}$ depend affinely on $\boldsymbol{s}$. Then, (32) represents a generalized eigenvalue problem (GEVP), which is a special type of SDP that can be solved efficiently, for example, with the bisection method [18]. We can simplify the additional optimization over the scalar variables ϵ_1 and ϵ_2 by noting the following:

Lemma 5: Let $\mathcal{S} = \big(\hat{T}_\mathrm{s}, \hat{\boldsymbol{Q}}_1, \hat{\boldsymbol{Q}}_2, \hat{\boldsymbol{Q}}_3, \hat{\boldsymbol{Z}}_1, \hat{\boldsymbol{Z}}_2, \hat{\boldsymbol{Z}}_3, \hat{\boldsymbol{R}}, \hat{\boldsymbol{Y}}, \hat{\epsilon}_1, \hat{\epsilon}_2\big)$ be an optimal solution to (32). Then, for any $c > 0$, $\mathcal{S}' = \big(\hat{T}_\mathrm{s}, \frac{1}{c}\hat{\boldsymbol{Q}}_1, \frac{1}{c}\hat{\boldsymbol{Q}}_2, \frac{1}{c}\hat{\boldsymbol{Q}}_3, \frac{1}{c}\hat{\boldsymbol{Z}}_1, \frac{1}{c}\hat{\boldsymbol{Z}}_2, \frac{1}{c}\hat{\boldsymbol{Z}}_3, \frac{1}{c}\hat{\boldsymbol{R}}, \frac{1}{c}\hat{\boldsymbol{Y}}, c\hat{\epsilon}_1, c\hat{\epsilon}_2\big)$ is also an optimal solution to (32).

Proof: Both solutions $\mathcal{S}$ and $\mathcal{S}'$ yield the same objective value $\frac{1}{\hat{T}_\mathrm{s}}$. To show that the feasibility of $\mathcal{S}$ implies the feasibility of $\mathcal{S}'$, we first consider the inequality constraint (25). If $\mathcal{S}$ is a feasible solution to (32), then it holds by applying the Schur complement that

$$\begin{bmatrix} 2\hat{\boldsymbol{Q}}_1 - \hat{\boldsymbol{R}} & \boldsymbol{0} & \hat{\boldsymbol{Y}}^\top\boldsymbol{B}^\mathsf{T} \\ * & \hat{\boldsymbol{Z}}_1 & \hat{\boldsymbol{Z}}_2 \\ * & * & \hat{\boldsymbol{Z}}_3 \end{bmatrix} - \frac{1}{\hat{\epsilon}_2}\begin{bmatrix} \boldsymbol{0} & \hat{\epsilon}_2\hat{\boldsymbol{Y}}^\top\boldsymbol{F}^\mathsf{T} \\ \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{H}^\mathsf{T} & \boldsymbol{0} \end{bmatrix}\begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{H}^\mathsf{T} \\ \hat{\epsilon}_2\boldsymbol{F}\hat{\boldsymbol{Y}}^\mathsf{T} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \succeq \boldsymbol{0}$$

$$\Leftrightarrow \begin{bmatrix} 2\frac{1}{c}\hat{\boldsymbol{Q}}_1 - \frac{1}{c}\hat{\boldsymbol{R}} & \boldsymbol{0} & \frac{1}{c}\hat{\boldsymbol{Y}}^\top\boldsymbol{B}^\mathsf{T} \\ * & \frac{1}{c}\hat{\boldsymbol{Z}}_1 & \frac{1}{c}\hat{\boldsymbol{Z}}_2 \\ * & * & \frac{1}{c}\hat{\boldsymbol{Z}}_3 \end{bmatrix} - \frac{1}{c\hat{\epsilon}_2}\begin{bmatrix} \boldsymbol{0} & c\hat{\epsilon}_2\frac{1}{c}\hat{\boldsymbol{Y}}^\top\boldsymbol{F}^\mathsf{T} \\ \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{H}^\mathsf{T} & \boldsymbol{0} \end{bmatrix}\begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{H}^\mathsf{T} \\ c\hat{\epsilon}_2\boldsymbol{F}\frac{1}{c}\hat{\boldsymbol{Y}}^\mathsf{T} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \succeq \boldsymbol{0},$$

where we have multiplied the inequality by $\frac{1}{c} > 0$. Consequently, $\mathcal{S}'$ also satisfies (25). We can proceed in a similar way for the constraint (24), which concludes the proof. ■

The two solutions $\mathcal{S}$ and $\mathcal{S}'$ also yield the same stabilizing control gain $\boldsymbol{K} = \boldsymbol{Y}(\boldsymbol{Q}_1)^{-1} = \frac{1}{c}\boldsymbol{Y}\big(\frac{1}{c}\boldsymbol{Q}_1\big)^{-1}$. As a consequence of Lemma 5, we can simplify the optimization problem (32) by setting $\epsilon_2 = \frac{1}{\epsilon_1}$.

Corollary 1: Solving the optimization problem (32) yields the same MCF and stabilizing control gain as solving

$$\begin{aligned} \min_{\substack{T_\mathrm{s},\boldsymbol{Q}_1,\boldsymbol{Q}_2,\\ \boldsymbol{Q}_3,\boldsymbol{Z}_1,\boldsymbol{Z}_2\\ \boldsymbol{Z}_3,\boldsymbol{R},\boldsymbol{Y},\epsilon}} \quad & \frac{1}{T_\mathrm{s}} \\ \text{s.t.} \quad & \left[\begin{array}{ccc|cc} & & & \boldsymbol{0} & \epsilon\left(\boldsymbol{Q}_1\boldsymbol{E}^\mathsf{T} + \boldsymbol{Y}^\mathsf{T}\boldsymbol{F}^\mathsf{T}\right) \\ & \boldsymbol{W}_{\hat{\boldsymbol{A}},\hat{\boldsymbol{B}}} & & \boldsymbol{H} & \boldsymbol{0} \\ & & & \boldsymbol{0} & \boldsymbol{0} \\ \hline * & * & * & -\epsilon\boldsymbol{I} & \boldsymbol{0} \\ * & * & * & * & -\epsilon\boldsymbol{I} \end{array}\right] \prec \boldsymbol{0}, \\ & \begin{bmatrix} 2\boldsymbol{Q}_1 - \boldsymbol{R} & \boldsymbol{0} & \boldsymbol{Y}^\mathsf{T}\hat{\boldsymbol{B}}^\mathsf{T} & \boldsymbol{0} & \epsilon\boldsymbol{Y}^\mathsf{T}\boldsymbol{F}^\mathsf{T} \\ * & \boldsymbol{Z}_1 & \boldsymbol{Z}_2 & \boldsymbol{0} & \boldsymbol{0} \\ * & * & \boldsymbol{Z}_3 & \boldsymbol{H} & \boldsymbol{0} \\ * & * & * & \epsilon\boldsymbol{I} & \boldsymbol{0} \\ * & * & * & * & \epsilon\boldsymbol{I} \end{bmatrix} \succeq \boldsymbol{0}, \\ & \boldsymbol{Q}_1 = \boldsymbol{Q}_1^\mathsf{T} \succ \boldsymbol{0}, \quad \boldsymbol{R} = \boldsymbol{R}^\mathsf{T} \succ \boldsymbol{0}, \quad \epsilon > 0. \end{aligned} \tag{33}$$

Proof: Let $\mathcal{S}$ be an optimal solution to (32), as defined in Lemma 5. We can make use of Lemma 5 and set $c = \frac{1}{\sqrt{\hat{\epsilon}_1\hat{\epsilon}_2}}$. Then, $\mathcal{S}'$ corresponds to the optimal solution of (33) and the result follows. ■

The reformulated optimization problem (33) has $6n^2 + nm + n + 2$ decision variables, which is one less than (32). In practice, having to perform only a scalar grid search over ϵ for solving (33) instead of a two-dimensional grid search over ϵ_1 and ϵ_2 for solving (32) significantly decreases the computational demand.

Remark 2: Analyzing stability via the time-delay approach [12] is advantageous: As long as the constraints in (33) hold, the sampling time can be changed online in $(0, T_{\mathrm{s,max}}]$ without losing stability guarantees, for example, to react to changes in the data transmission capacity.

C. Optimizing the Control Performance

Besides guaranteeing stability, we aim to optimize the control performance for a given sampling time $T_\mathrm{s} \in (0, T_{\mathrm{s,max}}]$. As a performance measure, we use the cost [19]

$$J = \int_{\tau_0}^{\tau_1} \tilde{\boldsymbol{x}}(t_k)^\mathsf{T}\boldsymbol{Q}_J\tilde{\boldsymbol{x}}(t_k) + \tilde{\boldsymbol{u}}(t)^\mathsf{T}\boldsymbol{R}_J\tilde{\boldsymbol{u}}(t)\mathrm{d}t_k, \tag{34}$$

where $\tau_1 - \tau_0 > 0$ is the optimization period, and $\boldsymbol{Q}_J \succ \boldsymbol{0}$, $\boldsymbol{R}_J \succ \boldsymbol{0}$ are weight matrices. To make the problem tractable,

we let $J \leq \bar{J} = \eta \int_{\tau_0}^{\tau_1} \tilde{\boldsymbol{x}}(t_k)^\mathsf{T} \boldsymbol{Q}_1^{-1} \boldsymbol{Q}_1^{-1} \tilde{\boldsymbol{x}}(t_k) \mathrm{d}t_k$ and consider the minimization of $\eta > 0$. By inserting the control law (22), substituting $\boldsymbol{K} = \boldsymbol{Y}\boldsymbol{Q}_1^{-1}$ and applying the Schur complement, we obtain that $J \leq \bar{J}$ for all $\tau_1 - \tau_0 > 0$ if

$$\begin{bmatrix} -\eta \boldsymbol{I} & \boldsymbol{Q}_1 & \boldsymbol{Y}^\mathsf{T} \\ * & -\boldsymbol{Q}_J^{-1} & \boldsymbol{0} \\ * & * & -\boldsymbol{R}_J^{-1} \end{bmatrix} \preceq \boldsymbol{0}. \tag{35}$$

Consequently, we can optimize the cost (34) while ensuring robust stability of the uncertain system (21) under the control (22) by solving the optimization problem

$$\begin{aligned} \min_{\substack{\eta, \boldsymbol{Q}_1, \boldsymbol{Q}_2, \boldsymbol{Q}_3, \boldsymbol{Z}_1, \\ \boldsymbol{Z}_2, \boldsymbol{Z}_3, \boldsymbol{R}, \boldsymbol{Y}, \epsilon_1, \epsilon_2}} \quad & \eta \\ \text{s.t.} \quad & \boldsymbol{Q}_1 = \boldsymbol{Q}_1^\mathsf{T} \succ \boldsymbol{0}, \quad \boldsymbol{R} = \boldsymbol{R}^\mathsf{T} \succ \boldsymbol{0}, \\ & (24), \quad (25), \quad (35), \end{aligned} \tag{36}$$

for a given sampling time $T_\mathrm{s} \in (0, T_{\mathrm{s,max}}]$. Note that (36) has a GEVP structure similar to (32). However, (36) does not allow for a simplification of the optimization over the scalar decision variables ϵ_1 and ϵ_2 since Lemma 5 does not apply due to the inequality constraint (35).

In this section, we have derived a method to calculate the MCF $f_{\mathrm{c,min}}$ required for robust control of (1) based on the uncertain linearized GP dynamics model (21). We can also compute a robustly stabilizing and optimal state-feedback controller (22) for a given control frequency $f_\mathrm{c} \geq f_{\mathrm{c,min}}$. Next, we evaluate our approach in simulation and study the tradeoff between data and the control frequency.

V. Evaluation

We consider a quadrotor flying in the vertical plane with position (x, z) and pitch angle θ. The system is simulated with the continuous-time dynamics [20]

$$\begin{aligned} m\ddot{x} &= -(T_1 + T_2)\sin(\theta) \\ m\ddot{z} &= (T_1 + T_2)\cos(\theta) - mg \\ I_{yy}\ddot{\theta} &= (T_1 - T_2)d, \end{aligned} \tag{37}$$

where $m = 0.1\,\mathrm{kg}$ is the mass, (T_1, T_2) are the motor thrusts, $g = 9.81\,\frac{\mathrm{m}}{\mathrm{s}^2}$ is the gravitational acceleration, $d = 0.1\,\mathrm{m}$ is the length of the effective moment arm of the propellers, and $I_{yy} = \frac{1}{12} m d^2$ is the inertia about the y-axis. By defining $\boldsymbol{x} = \left[x,\, \dot{x},\, z,\, \dot{z},\, \theta,\, \dot{\theta}\right]^\mathsf{T}$ and $\boldsymbol{u} = \left[T_1,\, T_2\right]^\mathsf{T}$, we can express (37) in the general form (1). We assume no prior knowledge about the dynamics, i.e., $\boldsymbol{f} \equiv \boldsymbol{0}$. We set the noise variance to $\boldsymbol{\Sigma}_\mathrm{n} = \mathrm{Diag}([0.1^2, \ldots, 0.1^2])$, and draw N training inputs uniformly from the set $\mathcal{Z} = \left\{(\boldsymbol{x}, \boldsymbol{u}) \in \mathbb{R}^8 \,|\, \underline{\boldsymbol{x}} \leq \boldsymbol{x} \leq \bar{\boldsymbol{x}},\, \underline{\boldsymbol{u}} \leq \boldsymbol{u} \leq \bar{\boldsymbol{u}}\right\}$, where $\underline{\boldsymbol{x}} = \left[0, -5\,\frac{\mathrm{m}}{\mathrm{s}}, 0, -5\,\frac{\mathrm{m}}{\mathrm{s}}, -\frac{\pi}{2}\,\mathrm{rad}, -5\,\frac{\mathrm{rad}}{\mathrm{s}}\right]^\mathsf{T}$, $\bar{\boldsymbol{x}} = \left[2\,\mathrm{m}, 5\,\frac{\mathrm{m}}{\mathrm{s}}, 2\,\mathrm{m}, 5\,\frac{\mathrm{m}}{\mathrm{s}}, \frac{\pi}{2}\,\mathrm{rad}, 5\,\frac{\mathrm{rad}}{\mathrm{s}}\right]^\mathsf{T}$, $\underline{\boldsymbol{u}} = \boldsymbol{0}$ and $\bar{\boldsymbol{u}} = \left[2\,\mathrm{N}, 2\,\mathrm{N}\right]^\mathsf{T}$. We set the threshold in Lemma 2 to $p^n = 0.99^6 \approx 0.94$. The SDPs (33) and (36) are solved using YALMIP [21] and MOSEK [22]. Additionally, we define the grid $\mathcal{G} = \{10^{-3}, 10^{-2.7}, \ldots, 10^{3}\}$ and optimize over $\epsilon \in \mathcal{G}$ in (33) and over $(\epsilon_1, \epsilon_2) \in \mathcal{G} \times \mathcal{G}$ in (36).

Fig. 2 shows the MCF computed by solving (33) for randomly drawn training sets of increasing size N, where ten different datasets are drawn for each value of N. We also provide the proportion of datasets for which (33) is feasible. We observe that a certain amount of data is required to stabilize the system and that the control frequency can be reduced by two-thirds when more data is available.

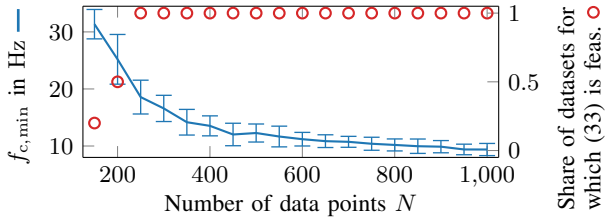

Fig. 2: Minimum control frequency required to ensure robust stability for different amounts of randomly drawn training data. The error bars represent $\pm$ one standard deviation. The circles show the proportion of datasets for which a stabilizing controller can be found by solving (33).

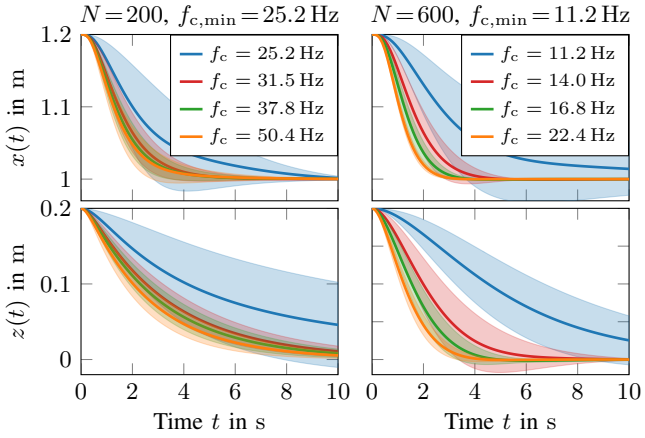

Fig. 3: Quadrotor trajectories for different control frequencies $f_\mathrm{c} = \xi f_{\mathrm{c,min}}$, where $\xi \in \{1, 1.25, 1.5, 2\}$, and different amounts of training data. The shaded areas represent $\pm$ one standard deviation. Convergence to the setpoint significantly improves, and variance reduces if the control frequency is increased from its minimum value $f_\mathrm{c} = f_{\mathrm{c,min}}$.

We also investigate the impact of the control frequency and model uncertainty on performance. For this, we set the desired operating point to $\boldsymbol{x}_\mathrm{e} = [1\,\mathrm{m}, 0, 0, 0, 0, 0]^\mathsf{T}$, $\boldsymbol{u}_\mathrm{e} = [0.4905\,\mathrm{N}, 0.4905\,\mathrm{N}]^\mathsf{T}$, the initial state to $\boldsymbol{x}(0) = [1.2\,\mathrm{m}, 0, 0.2\,\mathrm{m}, 0, 0, 0]^\mathsf{T}$ and the weight matrices in (34) to $\boldsymbol{Q}_J = \mathrm{diag}([100, 1, 100, 1, 100, 1])$, $\boldsymbol{R}_J = 0.01\boldsymbol{I}_2$. As discussed in Section IV-B, robust stability of the linearized system is guaranteed for all control frequencies $f_\mathrm{c} = \xi f_{\mathrm{c,min}}$ with $\xi \geq 1$. We evaluate $\xi \in \{1, 1.25, 1.5, 2\}$ with ten randomly drawn training sets each and compute the optimized controller by solving (36). Fig. 3 shows the quadrotor trajectories obtained from simulating the system for $10\,\mathrm{s}$ with mean and standard deviation for $N \in \{200, 600\}$. We observe the transient behavior improves significantly, and the variance reduces if the control frequency is increased from its minimum value.

For a systematic analysis of performance, we optimize for many frequencies $f_\mathrm{c} \in [10, 30]\,\mathrm{Hz}$ and simulate five different initial conditions. Fig. 4 shows the average LQR cost for a horizon of $10\,\mathrm{s}$ with the weight matrices $\boldsymbol{Q}_J$ and $\boldsymbol{R}_J$ and the contour lines of the cost. We observe a tradeoff between data

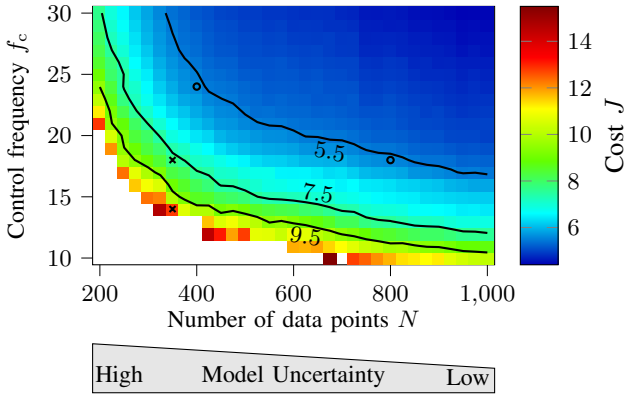

Fig. 4: Tradeoff between the control frequency and the amount of training data in terms of closed-loop performance. The white area indicates (N, f_c) pairs for which a stabilizing controller is found for less than 50% of the randomly drawn datasets. For an increasing amount of data, similar performance can be achieved at a lower control frequency. Vice versa, increasing the frequency can compensate for a significant lack of data.

and control frequency: As model uncertainty decreases due to more data, similar performance is achieved at a lower control frequency. On the other hand, for example, if we increase f_c by 33% from 18 Hz to 24 Hz, only half the amount of data ($N = 400$ instead of $N = 800$) is needed to get the same cost $J = 5.6$. Increasing the control frequency for a given amount of data reduces the cost, for example, by 42% when increasing f_c by 29% from 14 Hz to 18 Hz for $N = 350$. Furthermore, it is evident from the contour lines' shape that the sensitivity of the performance with respect to the control frequency increases with the size of the training data set.

VI. Discussion

Fig. 3 and Fig. 4 demonstrate that a higher control frequency improves performance and reduces variance. Considering this, the MCF provides a lower bound that allows us to safely reduce the control frequency, for example, to save computation or communication resources.

As illustrated in Fig. 2 and Fig. 4, the amount of data required for stability or achieving a specific performance depends on the frequency at which we can run the controller. A slight increase in control frequency can compensate for a significant lack of data, as is typical for physical systems such as robots, where data collection is expensive.

Stable regulation of the quadrotor is achieved for all simulated initial conditions, even when operating at the MCF, as shown in Fig. 3. This is remarkable, as the computation of the MCF via (33) is based on the linearized dynamics (21) and only considers the uncertainty corresponding to the GP variance, not the error due to linearizing the true nonlinear system (1). One reason is that the stability conditions in Theorem 1 are sufficient but not necessary conditions, and thus, the MCF includes some conservatism that may compensate for the linearization error, as is the case for our example.

VII. Conclusion

This work considers the control frequency as a design parameter for learning-based control of uncertain systems. To this end, we combine learning a continuous-time dynamics model using GPs with robust sampled-data control. This enables us to control the system at different frequencies without re-learning the model and to study the role of both the control frequency and the amount of data. We show that there is a tradeoff between the two design parameters in terms of stability and performance: Increasing the control frequency can make up for a lack of data and vice versa.

References

[1] F. Berkenkamp and A. P. Schoellig, "Safe and robust learning control with Gaussian processes," in *Proc. European Control Conference*, 2015, pp. 2496–2501.

[2] C. J. Ostafew, A. P. Schoellig, and T. D. Barfoot, "Robust constrained learning-based NMPC enabling reliable mobile robot path tracking," *International Journal of Robotics Research*, vol. 35, no. 13, pp. 1547–1563, 2016.

[3] A. von Rohr, M. Neumann-Brosig, and S. Trimpe, "Probabilistic robust linear quadratic regulators with Gaussian processes," in *Proc. Conf. on Learning for Dynamics and Control*, 2021, pp. 324–335.

[4] L. Brunke *et al.*, "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, pp. 411–444, 2022.

[5] A. Lederer, A. Capone, T. Beckers, J. Umlauft, and S. Hirche, "The impact of data on the stability of learning-based control," in *Proc. Conf. on Learning for Dynamics and Control*, 2021, pp. 623–635.

[6] D. Nguyen-Tuong and J. Peters, "Model learning for robot control: A survey," *Cognitive Processing*, vol. 12, pp. 319–340, 2011.

[7] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006.

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

[9] J. Berberich, S. Wildhagen, M. Hertneck, and F. Allgöwer, "Data-driven analysis and control of continuous-time systems under aperiodic sampling," *IFAC-PapersOnLine*, vol. 54, no. 7, pp. 210–215, 2021.

[10] X. Dai, A. Lederer, Z. Yang, and S. Hirche, "Can learning deteriorate control? Analyzing computational delays in Gaussian process-based event-triggered online learning," in *Proc. Conference on Learning for Dynamics and Control*, 2023, pp. 445–457.

[11] A. M. Metelli, F. Mazzolini, L. Bisi, L. Sabbioni, and M. Restelli, "Control frequency adaptation via action persistence in batch reinforcement learning," in *Proc. International Conference on Machine Learning*. PMLR, 2020, pp. 6862–6873.

[12] E. Fridman, *Introduction to time-delay systems: Analysis and control*. Springer, 2014.

[13] E. Fridman, A. Seuret, and J.-P. Richard, "Robust sampled-data stabilization of linear systems: An input delay approach," *Automatica*, vol. 40, no. 8, pp. 1441–1446, 2004.

[14] E. Fridman, "A refined input delay approach to sampled-data control," *Automatica*, vol. 46, no. 2, pp. 421–427, 2010.

[15] A. Seuret, "A novel stability analysis of linear systems under asynchronous samplings," *Automatica*, vol. 48, no. 1, pp. 177–182, 2012.

[16] A. Lederer, J. Umlauft, and S. Hirche, "Uniform error bounds for Gaussian process regression with application to safe control," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[17] L. Xie, "Output feedback H_∞ control of systems with parameter uncertainty," *International Journal of Control*, vol. 63, no. 4, pp. 741–750, 1996.

[18] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[19] H.-K. Lam and F. H. Leung, "Design and stabilization of sampled-data neural-network-based control systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 5, pp. 995–1005, 2006.

[20] Z. Yuan *et al.*, "safe-control-gym: A unified benchmark suite for safe learning-based control and reinforcement learning in robotics," *IEEE Robotics and Automation Let.*, vol. 7, no. 4, pp. 11 142–11 149, 2022.

[21] J. Lofberg, "Yalmip: A toolbox for modeling and optimization in matlab," in *Proc. IEEE International Conference on Robotics and Automation*, 2004, pp. 284–289.

[22] M. ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 10.0.*, 2022.