

# Stable Gaussian Process based Tracking Control of Euler-Lagrange Systems

Thomas Beckers<sup>a</sup>, Dana Kulić<sup>b</sup>, Sandra Hirche<sup>a</sup>

<sup>a</sup>*Chair of Information-oriented Control (ITR), Department of Electrical and Computer Engineering, Technical University of Munich, 80333 Munich, Germany*

<sup>b</sup>*Adaptive Systems Laboratory, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada*

---

## Abstract

Perfect tracking control for real-world Euler-Lagrange systems is challenging due to uncertainties in the system model and external disturbances. The magnitude of the tracking error can be reduced either by increasing the feedback gains or improving the model of the system. The latter is clearly preferable as it allows to maintain good tracking performance at low feedback gains. However, accurate models are often difficult to obtain.

In this article, we address the problem of high-performance tracking control for unknown Euler-Lagrange systems. In particular, we employ Gaussian Process Regression (GPR) to obtain a data-driven model that is used for the feed-forward compensation of unknown dynamics of the system. Beneficially, GPR provides not only an estimate of the uncertainties, but naturally provides a measure of model confidence depending on the distance to training points. Accordingly, the feedback gain is adapted based on the model fidelity allowing low feedback gains in state space regions of high model confidence. Additionally, we study the stability of GP-based tracking control for Euler-Lagrange systems. The proposed confidence-adaptive feedback control law guarantees a globally bounded tracking error with a specific probability. Simulation studies illustrate the results and demonstrate the superiority over state of the art tracking control approaches.

*Key words:* Stochastic control, Stability of nonlinear systems, Data-based control, Nonparametric methods, Adaptive system and control, robotic manipulators

---

## 1 Introduction

Euler-Lagrange systems represent a crucial and large class of dynamical systems, for which the equations of motion can be derived via the Euler-Lagrange equation. In the past decades, various control schemes for this class of systems have been proposed, most of them can be considered as a subset of computed torque control laws. With computed torque control, it is possible to derive very effective controllers that appear in robust, adaptive and learning control schemes [22]. The controller is separated into a feed-forward and a feedback part. A precise model of the true system is necessary to compensate the system dynamics to achieve a low gain feedback term.

This is beneficial in many ways: it avoids large errors in the presence of noise [12], avoids the saturation of actuators [13], and enhances safety in applications such as human-robot interaction [10]. Since the accuracy of the compensation depends on the precision of the model, all generalized external forces such as, e.g. in robotics, friction, payload or contact forces with the environment must be incorporated as precisely as possible.

However, an accurate model of these uncertainties is hard to obtain by classical first principles based techniques. Especially in modern applications of Lagrangian systems such as service robotics, the interaction with unstructured and a priori unknown environments further increases the uncertainty. A common approach is to derive a dynamic model from first order physics and increase the feedback gains of the control law to compensate the uncertainties until a desired tracking performance is achieved [24]. However, the increased gains are undesirable (as explained above) and building a more accurate model of the system for the entire workspace is

---

\* This paper was not presented at any IFAC meeting. Corresponding author T. Beckers. Tel. +49 (89) 289-25723. Fax +49 (89) 289-25724.

*Email addresses:* [t.beckers@tum.de](mailto:t.beckers@tum.de) (Thomas Beckers), [dana.kulic@uwaterloo.ca](mailto:dana.kulic@uwaterloo.ca) (Dana Kulić), [hirche@tum.de](mailto:hirche@tum.de) (Sandra Hirche).

time-consuming and effort demanding. Additionally, the stability of the closed loop system must be guaranteed to ensure safety and to prevent system destruction.

In this article, we address the problem of stable tracking control for Euler-Lagrange systems with unknown dynamics. For this purpose, we use Gaussian Process Regression (GPR) which is a promising, data-driven learning approach [11]. In particular, GPR is a supervised learning technique which combines several advantages. It requires only a minimum of prior knowledge to represent an arbitrary complex function, generalizes well even for small training data sets and has a precise trade-off between fitting the data and smoothing [19]. Since the prediction is based on Bayes' theorem, the Gaussian Process (GP) provides not only a mean function but also a predicted variance, and therefore a measure of the model fidelity based on the distance to the training data. We employ the provided model confidence to adapt the feedback gains in areas where it is necessary: i) to keep the system stable and ii) the tracking error less than a given value.

### 1.1 Related work

Computed torque control (CTC) requires a parametric model of the Euler-Lagrange system which can be identified, e.g. for robot manipulators, by taking advantage of the linearity in the parameters [21]. Errors in the identified parameters and unmodeled dynamics deteriorate the tracking performance and can affect the stability of the closed loop. Several methods are presented to overcome this problem, i.e. with online adaptation of the parametric model or varying feedback gains [23,20,15]. The drawback is that these approaches are based on an underlying parametric model which leads to the problem of model selection. The idea to use GPR as data-driven approach in control of robotic systems has been presented in [17,1]. However, no stability guarantees are given. Controller designs based on GPR with stability guarantees are still very scarce. In [8], a stable feedback linearization with online learning GPR is proposed. Numerical methods for determining the stability of GP forward models are given in [30]. Recently, also analytical results about the stability of dynamical systems with GPR are proposed [5]. Preliminary first results for stable control of Euler-Lagrange systems with GPR are presented by the authors in [3,4] but only for a restricted class of systems and without explicit determination of the tracking error. Thus, high performance tracking control for Euler-Lagrange systems with unknown dynamics and stability guarantees is still a largely open challenge.

### 1.2 Contributions

In this paper, we develop is a computed torque control law with GPR based feed-forward compensation. For

this purpose, a GP learns the unknown system dynamics from training data. If there is an estimated parametric model available, it is possible to include it as a priori knowledge. The contribution of this article is a control law that uses the mean of the GPR to compensate the unknown dynamics and the model confidence to adapt the gains. The derived method guarantees that the tracking error is uniformly ultimately bounded within a ball with a specific radius and a given probability. The applicability and efficacy of the proposed control approach is illustrated in simulation with a robot manipulator and a statistical study with randomly generated systems.

The remainder of the article starts with Section 2 where we introduce the class of Euler-Lagrange systems and GPR in general. Section 3 describes the training of the GP models and the computation of the model error. Section 4 contains the proposed control law and the proof of boundedness. The method is validated in Section 5 with numerical examples.

### 1.3 Notation

Vectors  $\mathbf{a}$  and vector-valued functions  $\mathbf{f}(\cdot)$  are denoted with bold characters. A Matrix  $A$  is described with capital letters. The identity matrix is given by  $I$ . Estimated values are indicated by a hat and errors by a tilde. The expression  $A_{:,i}$  denotes the  $i$ -th column of the matrix  $A$ . The expression  $\mathcal{N}(\mu, \Sigma)$  denotes a normal distribution with mean  $\mu$  and covariance  $\Sigma$ . The field of non-negative real numbers is denoted by  $\mathbb{R}_{\geq 0}$ , positive real numbers by  $\mathbb{R}_{> 0}$ . For natural numbers  $\mathbb{N}_{\geq 0}$ ,  $\mathbb{N}_{> 0}$  is used accordingly. The smallest eigenvalue of a matrix is denoted by  $\underline{\lambda}(\cdot)$  and the largest by  $\bar{\lambda}(\cdot)$ . The Euclidean norm is given by  $\|\cdot\|$  and the induced matrix norm of a matrix  $A$  by  $\|A\| = \bar{\lambda}(A^T A)^{1/2}$ . The expression  $\mathcal{H}_k$  denotes the reproducing kernel Hilbert space (RKHS) with the associated norm  $\|\cdot\|_k$  and  $\mathcal{C}$  the set of continuous functions.

## 2 Preliminaries and Definitions

### 2.1 Euler-Lagrange Systems

The class of Euler-Lagrange systems is defined by dynamical systems whose motion is described by the Euler-Lagrange equations. In this article, we restrict our focus on the class of non-conservative and fully-actuated systems where the set of equations is given by

$$\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathbf{u}_c + \mathbf{u}_d \quad (1)$$

with the generalized coordinates  $\mathbf{q} \in \mathbb{R}^n$  and the Lagrangian function  $\mathcal{L}: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$

$$\mathcal{L}(\dot{\mathbf{q}}, \mathbf{q}) := \mathcal{T}(\dot{\mathbf{q}}, \mathbf{q}) - \mathcal{V}(\mathbf{q}). \quad (2)$$

The Lagrangian function depends on the kinetic energy (or co-energy)  $\mathcal{T}: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  and the potential function  $\mathcal{V}: \mathbb{R}^n \rightarrow \mathbb{R}$ . Two types of generalized external forces are considered herein: The action of control  $\mathbf{u}_c \in \mathbb{R}^n$  and the effect of the unknown dynamics  $\mathbf{u}_d \in \mathbb{R}^n$ . We assume the following properties for the generalized forces.

**Assumption 1** *The generalized external forces can be parametrized as follows*

$$\mathbf{u}_c = \mathbf{u}(t), \quad \mathbf{u} \in \mathcal{C}, \mathbf{u}: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^n \quad (3)$$

$$\mathbf{u}_d = \mathbf{f}_u(\mathbf{p}), \quad \mathbf{f}_u \in \mathcal{C}, \mathbf{f}_u: \mathbb{R}^{3n} \rightarrow \mathbb{R}^n \quad (4)$$

with  $\mathbf{p} = [\dot{\mathbf{q}}^\top, \ddot{\mathbf{q}}^\top, \mathbf{q}^\top]^\top$  and the time  $t \in \mathbb{R}_{\geq 0}$ .

The assumption restricts the unknown dynamics  $\mathbf{f}_u$  to be state dependent which is, in most cases, a non restrictive condition. The kinetic energy is of the form

$$\mathcal{T}(\dot{\mathbf{q}}, \mathbf{q}) = \frac{1}{2} \dot{\mathbf{q}}^\top H(\mathbf{q}) \dot{\mathbf{q}}, \quad (5)$$

where  $H(\mathbf{q}) \in \mathbb{R}^{n \times n}$  is the generalized inertia matrix. Based on these assumptions, the Euler-Lagrange equations (1) can be written in the equivalent form

$$H(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) - \mathbf{f}_u(\mathbf{p}) = \mathbf{u}(t), \quad (6)$$

where the vector  $\mathbf{g}(\mathbf{q})$  is defined by

$$\mathbf{g}(\mathbf{q}) := \frac{\partial \mathcal{V}(\mathbf{q})}{\partial \mathbf{q}}. \quad (7)$$

The matrices  $H(\mathbf{q}), C(\mathbf{q}, \dot{\mathbf{q}}): \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ , and the vector  $\mathbf{g}(\mathbf{q}): \mathbb{R}^n \rightarrow \mathbb{R}^n$  have the following property:

**Property 1 (Structural Properties[18])** *The matrix  $H(\mathbf{q})$  is symmetric and positive definite and there is a special factorization of the matrix  $C(\mathbf{q}, \dot{\mathbf{q}})$ , so that  $\forall \dot{\mathbf{q}}, \mathbf{q} \in \mathbb{R}^n$*

- $\dot{H}(\mathbf{q}) = C(\mathbf{q}, \dot{\mathbf{q}}) + C(\mathbf{q}, \dot{\mathbf{q}})^\top \in \mathbb{R}^{n \times n}$  and thus,
- $\dot{H}(\mathbf{q}) - 2C(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$  is skew-symmetric.

## 2.2 Gaussian Process Regression

Let  $(\Omega, \mathcal{F}, P)$  be a probability space with the sample space  $\Omega = \mathbb{R}^n, n \in \mathbb{N}$ , the corresponding  $\sigma$ -algebra  $\mathcal{F}$  and the probability measure  $P$ . Assume a vector-valued, nonlinear function  $\mathbf{y} = \mathbf{f}(\mathbf{x})$  with  $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^n$  and  $\mathbf{y} \in \mathbb{R}^n$ . The measurement  $\tilde{\mathbf{y}} \in \mathbb{R}^n$  of the function is corrupted by Gaussian noise  $\boldsymbol{\eta} \in \mathbb{R}^n$ , i.e.

$$\tilde{\mathbf{y}} = \mathbf{f}(\mathbf{x}) + \boldsymbol{\eta} \quad (8)$$

$$\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \text{diag}(\sigma_1^2, \dots, \sigma_n^2)) \quad (9)$$

with the standard deviation  $\sigma_1, \dots, \sigma_n \in \mathbb{R}_{\geq 0}$ . To generate the training data, the function is evaluated at  $m$  input values  $\{\mathbf{x}^{(j)}\}_{j=1}^m$ . Together with the resulting measurements  $\{\tilde{\mathbf{y}}^{(j)}\}_{j=1}^m$ , the whole training data set is described by  $\mathcal{D} = \{X, Y\}$  with the input training matrix  $X = [\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(m)}] \in \mathbb{R}^{n \times m}$  and the output training matrix  $Y = [\tilde{\mathbf{y}}^{(1)}, \tilde{\mathbf{y}}^{(2)}, \dots, \tilde{\mathbf{y}}^{(m)}]^\top \in \mathbb{R}^{m \times n}$ . Now, the objective is to predict the output of the function  $\mathbf{y}^*$  at a test input  $\mathbf{x}^* \in \mathbb{R}^n$ .

The underlying assumption of Gaussian Process Regression is that the data can be represented as a sample of a multivariate Gaussian distribution. The joint distribution of the  $i$ -th component of  $\mathbf{y}^*$  is

$$\begin{bmatrix} Y_{:,i} \\ y_i^* \end{bmatrix} \sim \mathcal{N} \left( \mathbf{m}(\mathbf{x}), \begin{bmatrix} K_{\Phi_i}(X, X) & \mathbf{k}_{\Phi_i}(\mathbf{x}^*, X) \\ \mathbf{k}_{\Phi_i}(X, \mathbf{x}^*)^\top & k_{\Phi_i}(\mathbf{x}^*, \mathbf{x}^*) \end{bmatrix} \right) \quad (10)$$

with the covariance function  $k_{\Phi_i}: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  as a measure of the correlation of two points  $(\mathbf{x}, \mathbf{x}')$ . The function  $K_{\Phi_i}: \mathbb{R}^{n \times m} \times \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^{m \times m}$  is called the covariance matrix  $K_{j,l} = k_{\Phi_i}(X_{:,j}, X_{:,l})$  with  $j, l \in \{1, \dots, m\}$  where each element of the matrix represents the covariance between two elements of the training data  $X$ . The vector-valued covariance function  $\mathbf{k}_{\Phi_i}: \mathbb{R}^n \times \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^m$  calculates the covariance between the test input  $\mathbf{x}^*$  and the training data  $X$

$$\mathbf{k}_{\Phi_i}(\mathbf{x}^*, X) \text{ with } k_{\Phi_i,j} = k_{\Phi_i}(\mathbf{x}^*, X_{:,j}) \quad (11)$$

for all  $j \in \{1, \dots, m\}$  and  $i \in \{1, \dots, n\}$ . These functions depend on a set of hyperparameters  $\Phi_i = \{\varphi_i^{\{1\}}, \dots, \varphi_i^{\{n_h\}}\}$  whose number  $n_h \in \mathbb{N}$  and domain of parameters depends on the function used. The choice of the covariance function and the corresponding hyperparameters can be seen as degrees of freedom of the regression. A comparison of the characteristics of the different covariance functions can be found in [6]

The prediction of each component of  $\mathbf{y}^*$  is derived from the joint distribution (10) and therefore it is a Gaussian distributed variable. The conditional probability distribution is defined by the mean

$$\mu(y_i^* | \mathbf{x}^*, \mathcal{D}) = \mathbf{k}_{\Phi_i}(\mathbf{x}^*, X)^\top (K_{\Phi_i} + I\sigma_i^2)^{-1} Y_{:,i} \quad (12)$$

and the variance

$$\text{var}(y_i^* | \mathbf{x}^*, \mathcal{D}) = k_{\Phi_i}(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}_{\Phi_i}(\mathbf{x}^*, X)^\top (K_{\Phi_i} + I\sigma_i^2)^{-1} \mathbf{k}_{\Phi_i}(X, \mathbf{x}^*). \quad (13)$$

The  $n$  normally distributed components of  $\mathbf{y}^* | \mathbf{x}^*, \mathcal{D}$  are combined into a multi-variable Gaussian distribution

$$\begin{aligned} \mathbf{y}^* | \mathbf{x}^*, \mathcal{D} &\sim \mathcal{N}(\boldsymbol{\mu}(\cdot), \Sigma(\cdot)) \\ \boldsymbol{\mu}(\mathbf{y}^* | \mathbf{x}^*, \mathcal{D}) &= [\mu(y_1^* | \mathbf{x}^*, \mathcal{D}), \dots, \mu(y_n^* | \mathbf{x}^*, \mathcal{D})]^\top \\ \Sigma(\mathbf{y}^* | \mathbf{x}^*, \mathcal{D}) &= \text{diag}(\text{var}(y_1^* | \mathbf{x}^*, \mathcal{D}), \dots, \text{var}(y_n^* | \mathbf{x}^*, \mathcal{D})). \end{aligned} \quad (14)$$

where the hyperparameters  $\varphi_i^{\{1\}}, \dots, \varphi_i^{\{n_h\}} \in \Phi_i$  are optimized by means of the likelihood function, thus by

$$\varphi_i^{\{j\}} = \arg \max_{\varphi_i^{\{j\}}} \log P(Y_{:,i}|X, \varphi_i^{\{j\}}) \quad (15)$$

for all  $i \in \{1, \dots, n\}$  and  $j \in \{1, \dots, n_h\}$ . For this purpose, a gradient based algorithm is often used to find a (local) maximum of the likelihood function [19].

### 2.2.1 Marginal Variance

The computation of the variance with respect to a subset of elements of  $\mathbf{x}^*$  can be done by marginalization. Assume  $\mathbf{x}^* = [\mathbf{x}_1^{*\top}, \mathbf{x}_2^{*\top}]^\top$  with  $\mathbf{x}_1^* \in \mathbb{R}^{n_1}$ ,  $\mathbf{x}_2^* \in \mathbb{R}^{n_2}$  and  $\mathbf{x}^* \in \mathbb{R}^{n=n_1+n_2}$ . Then, the marginal variance of the prediction based on  $\mathbf{x}_1^*$  is given by

$$\begin{aligned} \text{var}(y_i^*|\mathbf{x}_1^*, \mathcal{D}) &= k_{\tilde{\Phi}_i}(\mathbf{x}_1^*, \mathbf{x}_1^*) - \mathbf{k}_{\tilde{\Phi}_i}(\mathbf{x}_1^*, X_{1:n_1,:})^\top \\ &\quad (K_{\tilde{\Phi}_i}(X_{1:n_1,:}, X_{1:n_1,:}) + I\sigma_i^2)^{-1} \\ &\quad \mathbf{k}_{\tilde{\Phi}_i}(\mathbf{x}_1^*, X_{1:n_1,:}) \end{aligned} \quad (16)$$

with the necessary subset of hyperparameters  $\tilde{\Phi}_i \subset \Phi_i$  for the covariance function defined on the input space  $\mathbb{R}^{n_1}$ . The variable  $X_{1:n_1,:}$  denotes a subset of the input data, i.e. the first  $n_1$  rows. Thus, the combined marginal variance is rewritten as

$$\Sigma(\mathbf{y}^*|\mathbf{x}_1^*, \mathcal{D}) = \text{diag}(\text{var}_1(\cdot), \dots, \text{var}_{n_1}(\cdot)). \quad (17)$$

## 3 Gaussian Process Model

In this section, we introduce the learning procedure and the computation of the model error, i.e. the error between the true system and the modeled dynamics.

### 3.1 Learning

Consider the Euler-Lagrange system in (6) with the unknown residual dynamics  $\mathbf{f}_u$ . If a priori knowledge of the plant is available, a hybrid learning approach can be used which is a combination of a parametric and a data-driven model. We consider the parametric model is given by the Euler-Lagrange dynamics

$$\hat{\mathbf{u}}(t) = \hat{H}(\mathbf{q})\ddot{\mathbf{q}} + \hat{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \hat{\mathbf{g}}(\mathbf{q}) \quad (18)$$

where  $\hat{H} \in \mathbb{R}^{n \times n}$ ,  $\hat{C} \in \mathbb{R}^{n \times n}$  and  $\hat{\mathbf{g}} \in \mathbb{R}^n$  are estimates of the true matrices which also satisfy Property 1. Furthermore, the estimates must fulfill the following property.

**Property 2 (Structure of the estimates)** *There exist constants  $h_1, h_2, k_C \in \mathbb{R}_{>0}$  such that  $h_1\|\mathbf{x}\|^2 \leq \mathbf{x}^\top \hat{H}(\mathbf{q})\mathbf{x} \leq h_2\|\mathbf{x}\|^2$ ,  $\|\hat{C}(\mathbf{q}, \dot{\mathbf{q}})\| \leq k_C\|\dot{\mathbf{q}}\|$ , and  $\hat{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{q}' = \hat{C}(\mathbf{q}, \mathbf{q}')\dot{\mathbf{q}}$  for all  $\mathbf{q}, \dot{\mathbf{q}}, \mathbf{q}', \mathbf{x} \in \mathbb{R}^n$ .*

Thus, the matrix  $\hat{H}$  is bounded and the matrix  $\hat{C}(\mathbf{q}, \dot{\mathbf{q}})$  is bounded in  $\mathbf{q}$  and linear in  $\dot{\mathbf{q}}$ . The identification of these estimates while guaranteeing Properties 1 and 2 can be achieved following the identification procedures from [25,14]. Please note that Property 2 is required for the estimates only and not for the true system matrices of (6).

**Remark 1** *Without prior knowledge of the system, the estimates are set to  $\hat{H} = I, \hat{C} = 0, \hat{\mathbf{g}} = \mathbf{0}$ .*

After the parametric model is selected, a Gaussian Process is trained with  $m$  data pairs  $\mathcal{D} = \{\mathbf{p}^{\{i\}}, \tilde{\boldsymbol{\tau}}^{\{i\}}\}_{i=1}^m$  which consist of

$$\begin{aligned} \mathbf{p} &= [\ddot{\mathbf{q}}^\top, \dot{\mathbf{q}}^\top, \mathbf{q}^\top]^\top \in \mathbb{R}^{3n} \\ \tilde{\boldsymbol{\tau}}(\mathbf{p}) &= \tilde{H}(\mathbf{q})\ddot{\mathbf{q}} + \tilde{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \tilde{\mathbf{g}}(\mathbf{q}) - \mathbf{f}_u(\mathbf{p}), \end{aligned} \quad (19)$$

with  $\tilde{H} = H - \hat{H}$ ,  $\tilde{C} = C - \hat{C}$  and  $\tilde{\mathbf{g}} = \mathbf{g} - \hat{\mathbf{g}}$ . For the generation of training data, the system (6) can be actuated by an arbitrary controller as shown in Fig. 1. The only condition is that the closed loop system behave well enough to generate a finite sequence of training data. Stability is not necessarily required as long as the closed-loop system states remain bounded for the finite training time.

### 3.2 Model error

For the computation of the model error, we assume the following for the covariance function of the Gaussian Process.

**Assumption 2** *The covariance function  $k(\cdot, \cdot)$  is chosen such that the function  $\tilde{\boldsymbol{\tau}}(\mathbf{p}) = [\tilde{\tau}_1(\mathbf{p}), \dots, \tilde{\tau}_n(\mathbf{p})]^\top$  has a bounded RKHS norm on any closed set  $D \subset \mathbb{R}^{3n}$ , i.e.  $\|\tilde{\tau}_i\|_k < \infty$  must be satisfied for all  $i = 1, \dots, n$ .*

**Remark 2** *The norm of a function  $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^n$  in a RKHS is a smoothness measure relative to a covariance function  $k(x, x')$  that is uniquely connected with this*

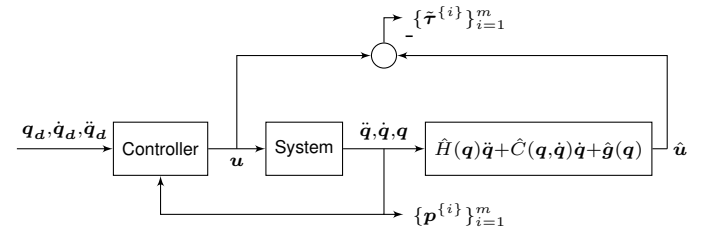


Fig. 1. Diagram for the generation of the training data set  $\mathcal{D} = \{\mathbf{p}^{\{i\}}, \tilde{\boldsymbol{\tau}}^{\{i\}}\}_{i=1}^m$ .

RKHS. In particular, it is a Lipschitz constant

$$\frac{|f_i(\mathbf{x}) - f_i(\mathbf{x}')|}{d_k(\mathbf{x}, \mathbf{x}')} \leq \|f_i\|_k, \quad \forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^n \quad (20)$$

with respect to the metric of the used covariance function  $d_k : \mathbb{R}^n \times \mathbb{R}^n \rightarrow [0, \infty)$  given by

$$d_k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}) + k(\mathbf{x}', \mathbf{x}') - 2k(\mathbf{x}, \mathbf{x}'). \quad (21)$$

A more detailed discussion about RKHS norms is given in [31].

Assumption 2 requires that the covariance function must be selected in such a way that the residual  $\tilde{\tau}(\mathbf{p})$  is an element of the associated RKHS. This sounds paradoxical since the residual is unknown. However, there exist some covariance functions, so called universal functions, which can approximate any continuous function arbitrarily precisely on a compact set [27, Lemma 4.55]. Therefore, many residual dynamics can be covered by the universal covariance function so that this assumption is not restrictive. Additionally, [7] presents a covariance function which is very effective in learning the inverse dynamics of Euler-Lagrange system.

An upper bound for the distance between the mean prediction  $\boldsymbol{\mu}(\tilde{\tau})$  of the Gaussian Process Regression and the true function is given in [26] and is extended for multi-dimensional functions in the following lemma.

**Lemma 1** Consider a Lagrangian system (6) and a trained Gaussian Process satisfying Assumption 2. The model error is bounded by

$$P\left\{\|\boldsymbol{\mu}(\tilde{\tau}|\mathbf{p}, \mathcal{D}) - \tilde{\tau}(\mathbf{p})\| \leq \|\boldsymbol{\beta}^\top \Sigma^{\frac{1}{2}}(\tilde{\tau}|\mathbf{p}, \mathcal{D})\|\right\} \geq (1 - \delta)^n \quad (22)$$

for  $\mathbf{p} \in D$  with  $\delta \in (0, 1)$ ,  $\boldsymbol{\beta} \in \mathbb{R}^n$  and

$$\beta_j = \sqrt{2\|\tilde{\tau}_j\|_k^2 + 300\gamma_j \ln^3\left(\frac{m+1}{\delta}\right)} \quad (23)$$

$$\gamma_j = \max_{\mathbf{p}^{\{1\}}, \dots, \mathbf{p}^{\{m+1\}} \in D} \frac{1}{2} \log |I + \sigma_i^{-2} K_{\Phi_j}(\mathbf{x}, \mathbf{x}')|. \quad (24)$$

$$\mathbf{x}, \mathbf{x}' \in \{\mathbf{p}^{\{1\}}, \dots, \mathbf{p}^{\{m+1\}}\} \quad (25)$$

**PROOF.** See appendix.

**Remark 3** If Assumption 2 is not fulfilled due to the wrong choice of covariance function or hyperparameters, for many common covariance functions the model error is still bounded on a closed set [2]. However, this may result in looser upper bounds for the model error.

The information capacity  $\gamma$  has a sub-linear dependency on the number of training points for many commonly used covariance functions and can be approximated with a constant, e.g. shown in [26]. Therefore, even though the values of the elements of  $\boldsymbol{\beta}$  are increasing with the number of training data, it is possible to learn the true function  $\tilde{\tau}(\mathbf{p})$  arbitrarily exactly [5]. The result of Lemma 1 is an upper bound for the model error. The stochastic nature of the bound is due to the fact that just a finite number of noisy training points are available and thus, the true function cannot be known exactly. If exact knowledge of the model was available, the variance of the GPR would be zero and thus, the upper bound for the model error would also be zero. A simple example is shown in Fig. 2 that illustrates the Lemma. Here we assume, for the sake of simplification, that  $\mathbf{p}$  is just one dimensional and  $\tilde{\tau}$  is an element of a RKHS defined by the linear covariance function  $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{x}'$ . Lemma 1 shows that the true function remains inside

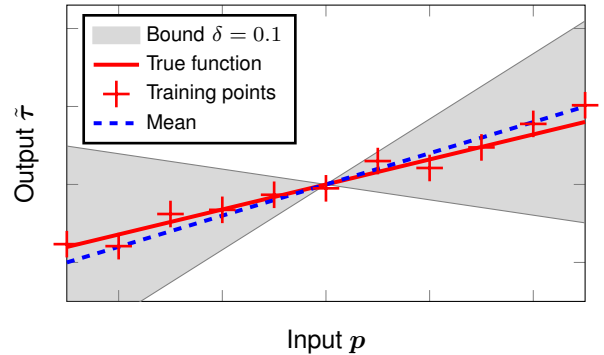


Fig. 2. The model error  $\Delta$  is the difference between the true function (dashed blue) and the mean function (solid red) of the GPR. Regarding to Lemma 1, the true function remains inside the shaded area with a probability of 90%

the shaded area around the mean function of the GPR with  $\delta = 0.1$ , i.e. a probability of 90%. With an increasing number of training points or decreasing noise  $\sigma$  of the training data, the bound becomes tighter. Since the model is used for a feed-forward compensation of the unknown dynamics of the system, the model error directly affects the tracking error as shown in the next section.

#### 4 Tracking control with GPR

The goal of tracking control is to follow a desired trajectory with the closed loop system. A common approach for tracking control of Euler-Lagrange systems is computed torque control, which contains a feed-forward and a feedback part. In this setting, low feedback gains are desirable because of several advantages such as a good noise attenuation. However, due to the influence of the unknown dynamics, the feedback gains must be greater than a specific value to keep the tracking error under a predefined limit. We use the mean of the Gaussian Process Regression to feed-forward compensate the residual

dynamics  $\tilde{\tau}(\mathbf{p})$  and adapt the feedback gains based on the model fidelity. For this purpose, the uncertainty of the regression is used to scale the feedback gains. We start with the following assumption for the desired trajectory.

**Assumption 3** *The desired state trajectory is bounded by  $\|\mathbf{q}_d\| \leq \bar{q}_d$ ,  $\|\dot{\mathbf{q}}_d\| \leq \bar{\dot{q}}_d$  with  $\bar{q}_d, \bar{\dot{q}}_d \in \mathbb{R}_{\geq 0}$ ,  $\mathbf{q}_d \in \mathbb{R}^n$ .*

Assumption 3, i.e. bounded reference motion trajectories, is a very natural assumption and does not pose any restriction in practice. Before the control law is proposed, the following lemma is introduced.

**Lemma 2** *Let  $\Sigma_d : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  and  $\Sigma_p : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  be the marginal variances which are defined analogously to (16) by*

$$\begin{aligned}\Sigma_d(\dot{\mathbf{q}}, \mathbf{q}) &:= \Sigma(\tilde{\tau}|\dot{\mathbf{q}}, \mathbf{q}, \mathcal{D}) \\ \Sigma_p(\mathbf{q}) &:= \Sigma(\tilde{\tau}|\mathbf{q}, \mathcal{D}).\end{aligned}\quad (26)$$

i) *Let  $K_d, K_p : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n \times n}$  be positive definite, symmetric matrix functions such that  $(K_d \circ \Sigma_d)$ ,  $(K_p \circ \Sigma_p)$  are continuous and bounded by*

$$k_{d1}\|\mathbf{x}\|^2 \leq \mathbf{x}^\top K_d(\Sigma_d(\dot{\mathbf{q}}, \mathbf{q}))\mathbf{x} \leq k_{d2}\|\mathbf{x}\|^2 \quad (27)$$

$$k_{p1}\|\mathbf{x}\|^2 \leq \mathbf{x}^\top K_p(\Sigma_p(\mathbf{q}))\mathbf{x} \leq k_{p2}\|\mathbf{x}\|^2, \quad (28)$$

for all  $\dot{\mathbf{q}}, \mathbf{q}, \mathbf{x} \in \mathbb{R}^n$  with  $k_{p1}, k_{p2}, k_{d1}, k_{d2} \in \mathbb{R}_{>0}$ . Then, there exists an  $\epsilon > 0$  such that the matrix  $M \in \mathbb{R}^{2n \times 2n}$  given by<sup>1</sup>

$$M = \begin{bmatrix} \underbrace{-K_d(\Sigma_d) + \epsilon \hat{H}}_{M_{11} \in \mathbb{R}^{n \times n}} & \underbrace{\frac{\epsilon}{2}(-K_d^\top(\Sigma_d) + \hat{C})}_{M_{12} \in \mathbb{R}^{n \times n}} \\ \underbrace{\frac{\epsilon}{2}(-K_d(\Sigma_d) + \hat{C}^\top)}_{M_{12} \in \mathbb{R}^{n \times n}} & \underbrace{-\epsilon K_p(\Sigma_p)}_{M_{22} \in \mathbb{R}^{n \times n}} \end{bmatrix} \quad (29)$$

is negative definite for all  $\dot{\mathbf{q}}, \mathbf{q} \in \mathbb{R}^n$ .

**PROOF.** See appendix.

The next theorem introduces the control law with guaranteed boundedness of the tracking error.

**Theorem 1 (CTC-GPR)** *Consider the Lagrangian system (6) and a Gaussian Process trained with (19) which satisfies Assumptions 1 and 2. Let  $\mathbf{e} = \mathbf{q} - \mathbf{q}_d$ ,  $\dot{\mathbf{e}} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_d$  be the tracking error. The control law*

$$\begin{aligned}\mathbf{u}(t) &= \hat{H}(\mathbf{q})\ddot{\mathbf{q}}_d + \hat{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}_d + \hat{\mathbf{g}}(\mathbf{q}) + \boldsymbol{\mu}(\tilde{\tau}|\mathbf{p}, \mathcal{D}) \\ &\quad - K_d(\Sigma_d)\dot{\mathbf{e}} - K_p(\Sigma_p)\mathbf{e}\end{aligned}\quad (30)$$

<sup>1</sup> For notational convenience, the dependencies of  $H, C, \mathbf{g}$  and  $\Sigma_d, \Sigma_p$  are dropped here.

with Assumption 3 and Lemma 2 guarantees that there exist a closed set  $D$  and a model error  $\hat{\Delta}$  such that

$$P \left\{ \left\| \begin{array}{l} \dot{\mathbf{e}}(t) \\ \mathbf{e}(t) \end{array} \right\| \leq r, \forall t \geq t_0 + T(\delta) \right\} \geq (1 - \delta)^n \quad (31)$$

for any initial value  $\|\dot{\mathbf{e}}_0^\top(t_0), \mathbf{e}^\top(t_0)\| < \delta$  with  $t_0, T(\delta), \delta, r \in \mathbb{R}_{>0}$ .

Before proving the theorem we provide a series of results on a suitable Lyapunov candidate.

**Lemma 3** *There exist an  $\epsilon > 0$  such that*

$$V = \frac{1}{2}\dot{\mathbf{e}}^\top \hat{H}(\mathbf{q})\dot{\mathbf{e}} + \int_0^e \mathbf{z}^\top K_p(\Sigma_p(\mathbf{z} + \mathbf{q}_d))d\mathbf{z} + \epsilon \mathbf{e}^\top \hat{H}(\mathbf{q})\dot{\mathbf{e}} \quad (32)$$

is a radially unbounded Lyapunov function.

**PROOF.** To ensure that the Lyapunov candidate is positive definite, the domain of the integral in (32) is analyzed. For this purpose, we define an additional matrix function  $\underline{K}_p : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n \times n}$  with

$$\underline{K}_p(\Sigma_p) = I \min_{i=\{1, \dots, n\}} \lambda_i(K_p(\Sigma_p)), \quad (33)$$

which is continuous because  $K_p$  is a continuous function. Then, the integral is lower bounded by

$$\begin{aligned}\int_0^e \mathbf{z}^\top K_p(\Sigma_p)d\mathbf{z} &\geq \int_0^e \mathbf{z}^\top \underline{K}_p(\Sigma_p)d\mathbf{z} \\ &= \sum_{i=1}^n \int_0^{e_i} z_i \min_{i=\{1, \dots, n\}} \lambda_i(K_p(\Sigma_p))dz_i \\ &\geq \frac{1}{2} \sum_{i=1}^n \min_{\mathbf{q} \in \mathbb{R}, i=\{1, \dots, n\}} \lambda_i(K_p(\Sigma_p(\mathbf{q})))e_i^2 \\ &\geq \frac{1}{2}k_{p1}\|\mathbf{e}\|^2.\end{aligned}\quad (34)$$

An upper quadratic bound can be found in an analogous way using the maximum eigenvalue of  $K_p$ . Since the integral is lower bounded and  $\hat{H}(\mathbf{q})$  is always positive definite, the parameter  $\epsilon$  can be chosen sufficiently small to achieve a positive definite and radially unbounded Lyapunov function. The valid interval for  $\epsilon$  can be determined by the lower bound of the Lyapunov function (32)

$$\dot{V}(\dot{\mathbf{e}}, \mathbf{e}) \geq \frac{1}{2}h_1\|\dot{\mathbf{e}}\|^2 + \frac{1}{2}k_{p1}\|\mathbf{e}\|^2 - \frac{1}{2}\epsilon h_2 \left( \|\dot{\mathbf{e}}\|^2 + \|\mathbf{e}\|^2 \right) \quad (35)$$

which is positive for

$$0 < \varepsilon < \min \left\{ \frac{k_{p1}}{h_2}, \frac{h_1}{h_2} \right\}. \quad \square \quad (36)$$

In the next step, we derive an upper bound for the time derivative of the Lyapunov function.

**Lemma 4** *Assume the Lyapunov function (32) and the system (6) with the control law (30). The drift of (32) is upper bounded by*

$$\begin{aligned} \dot{V}(\dot{\mathbf{e}}, \mathbf{e}) &\leq -\frac{3}{4}v_1\|\dot{\mathbf{e}}\|^2 - \frac{3}{4}\varepsilon v_2\|\mathbf{e}\| + \varepsilon k_C\|\dot{\mathbf{e}}\|^2\|\mathbf{e}\| \\ &\quad + \frac{\bar{\Delta}^2}{v_1} + \varepsilon \frac{\bar{\Delta}^2}{v_2}. \end{aligned} \quad (37)$$

with  $v_1, v_2, \bar{\Delta} \in \mathbb{R}_{>0}$  and  $\mathbf{p} \in D$ .

**PROOF.** The time derivative of (32) is expressed by

$$\dot{V}(\dot{\mathbf{e}}, \mathbf{e}) = \begin{bmatrix} \dot{\mathbf{e}}^\top \hat{H} + \varepsilon \mathbf{e}^\top \hat{H} \\ \mathbf{e}^\top K_p(\Sigma_p) + \frac{1}{2} \dot{\mathbf{e}}^\top \hat{H} + \varepsilon (\mathbf{e}^\top \hat{H} + \dot{\mathbf{e}}^\top \hat{H}) \end{bmatrix}^\top \begin{bmatrix} \dot{\mathbf{e}} \\ \mathbf{e} \end{bmatrix}. \quad (38)$$

With Property 1 and (30), it can be rewritten as

$$\begin{aligned} \dot{V} &= \begin{bmatrix} \dot{\mathbf{e}} \\ \mathbf{e} \end{bmatrix}^\top \underbrace{\begin{bmatrix} -K_d(\Sigma_d) + \varepsilon \hat{H} & \frac{\varepsilon}{2}(-K_d^\top(\Sigma_d) + \hat{C}) \\ \frac{\varepsilon}{2}(-K_d(\Sigma_d) + \hat{C}^\top) & -\varepsilon K_p(\Sigma_p) \end{bmatrix}}_{M \in \mathbb{R}^{2n \times 2n}} \begin{bmatrix} \dot{\mathbf{e}} \\ \mathbf{e} \end{bmatrix} \\ &\quad + (\dot{\mathbf{e}} + \varepsilon \mathbf{e})^\top (\boldsymbol{\mu}(\tilde{\boldsymbol{\tau}}|\mathbf{p}, \mathcal{D}) - \tilde{\boldsymbol{\tau}}(\mathbf{p})). \end{aligned} \quad (39)$$

For the analysis, we compute bounds for the elements of  $M$  to find an upper bound for the drift of the Lyapunov function. The following statements can be made for the matrix  $M$ : The submatrix  $M_{11} \in \mathbb{R}^{n \times n}$  is negative definite since Lemma 2 guarantees the negative definiteness of  $-K_d(\Sigma_d)$  and the parameter  $\varepsilon$  can be chosen sufficiently small. In addition, the submatrix is bounded with

$$\dot{\mathbf{e}}^\top M_{11} \dot{\mathbf{e}} = \dot{\mathbf{e}}^\top (-K_d(\Sigma_d) + \varepsilon H) \dot{\mathbf{e}} \leq (-k_{d1} + \varepsilon h_2) \|\dot{\mathbf{e}}\|^2. \quad (40)$$

Lemma 2 ensures the negative definiteness of the submatrix  $M_{22} \in \mathbb{R}^{n \times n}$  with  $\mathbf{e}^\top M_{22} \mathbf{e} \leq -\varepsilon k_{p1} \|\mathbf{e}\|^2$ . With Assumption 3 and Property 2, the submatrix  $M_{12} \in \mathbb{R}^{n \times n}$  is bounded by

$$\mathbf{e}^\top M_{12} \dot{\mathbf{e}} \leq \varepsilon (k_C \|\dot{\mathbf{q}}_d - \dot{\mathbf{e}}\| + k_{d2}) \|\dot{\mathbf{e}}\| \|\mathbf{e}\| \quad (41)$$

$$\leq \varepsilon (k_C \|\dot{\mathbf{e}}\| + k_C \bar{q}_d + k_{d2}) \|\dot{\mathbf{e}}\| \|\mathbf{e}\| \quad (42)$$

With Lemma 1, the overall upper bound for the time derivative of the Lyapunov function is given by

$$\begin{aligned} \dot{V}(\dot{\mathbf{e}}, \mathbf{e}) &\leq (\varepsilon h_2 - k_{d1}) \|\dot{\mathbf{e}}\|^2 - \varepsilon k_{p1} \|\mathbf{e}\|^2 \\ &\quad + \varepsilon (k_C \|\dot{\mathbf{e}}\| + k_C \bar{q}_d + k_{d2}) \|\dot{\mathbf{e}}\| \|\mathbf{e}\| \\ &\quad + (\|\dot{\mathbf{e}}\| + \varepsilon \|\mathbf{e}\|) \|\boldsymbol{\beta}^\top \Sigma(\tilde{\boldsymbol{\tau}}|\mathbf{p}, \mathcal{D})\| \end{aligned} \quad (43)$$

Considering the inequality

$$\|\dot{\mathbf{e}}\| \|\mathbf{e}\| \leq \frac{1}{2} \left( \rho \|\dot{\mathbf{e}}\|^2 + \frac{\mathbf{e}^2}{\rho} \right) \quad (44)$$

that holds for all  $\dot{\mathbf{e}}, \mathbf{e} \in \mathbb{R}^n$  and  $\rho \in \mathbb{R}_{\geq 0}$ , (43) can be rewritten as

$$\begin{aligned} \dot{V}(\dot{\mathbf{e}}, \mathbf{e}) &\leq (\varepsilon h_2 - k_{d1}) \|\dot{\mathbf{e}}\|^2 - \varepsilon k_{p1} \|\mathbf{e}\|^2 \\ &\quad + \frac{\varepsilon}{2} (k_C \bar{q}_d + k_{d2}) (\rho \|\dot{\mathbf{e}}\|^2 + \frac{\|\mathbf{e}\|^2}{\rho} + \varepsilon k_C \|\dot{\mathbf{e}}\|^2 \|\mathbf{e}\| \\ &\quad + (\|\dot{\mathbf{e}}\| + \varepsilon \|\mathbf{e}\|) \|\boldsymbol{\beta}^\top \Sigma(\tilde{\boldsymbol{\tau}}|\mathbf{p}, \mathcal{D})\| \end{aligned} \quad (45)$$

with  $\rho = (1 + \varepsilon_2) \frac{k_C \bar{q}_d + k_{d2}}{2k_{p1}}$ ,  $\varepsilon_2 \in \mathbb{R}_{>0}$ .

The choice of  $\rho$  guarantees that the factors of the quadratic parts are still negative:

$$\begin{aligned} \dot{V}(\dot{\mathbf{e}}, \mathbf{e}) &\leq \left( \varepsilon h_2 - k_{d1} + \frac{\varepsilon \rho}{2} (k_C \bar{q}_d + k_{d2}) \right) \|\dot{\mathbf{e}}\|^2 \\ &\quad - \varepsilon k_{p1} \frac{\varepsilon_2}{1 + \varepsilon_2} \|\mathbf{e}\|^2 + \varepsilon k_C \|\dot{\mathbf{e}}\|^2 \|\mathbf{e}\| \\ &\quad + (\|\dot{\mathbf{e}}\| + \varepsilon \|\mathbf{e}\|) \|\boldsymbol{\beta}^\top \Sigma(\tilde{\boldsymbol{\tau}}|\mathbf{p}, \mathcal{D})\| \end{aligned} \quad (46)$$

With the inequality

$$v_1 \|\mathbf{x}\| \leq \frac{v_1^2}{v_2} + \frac{v_2}{4} \|\mathbf{x}\|^2 \quad (47)$$

that holds for all  $\mathbf{x} \in \mathbb{R}^n$  and  $v_1, v_2 \in \mathbb{R}_{\geq 0}$  the linear part of (46) can be bounded by a quadratic function

$$\begin{aligned} &(\|\dot{\mathbf{e}}\| + \varepsilon \|\mathbf{e}\|) \|\boldsymbol{\beta}^\top \Sigma(\tilde{\boldsymbol{\tau}}|\mathbf{p}, \mathcal{D})\| \\ &\leq \frac{\bar{\Delta}^2}{v_1} + \frac{v_1}{4} \|\dot{\mathbf{e}}\|^2 + \frac{\varepsilon^2 \bar{\Delta}^2}{\varepsilon v_2} + \frac{\varepsilon v_2}{4} \|\mathbf{e}\|^2 \end{aligned} \quad (48)$$

with  $\bar{\Delta} \in \mathbb{R}_{>0}$  and

$$\bar{\Delta} \geq \|\boldsymbol{\beta}^\top \Sigma(\tilde{\boldsymbol{\tau}}|\mathbf{p}, \mathcal{D})\| \quad (49)$$

$$v_1 := -\varepsilon h_2 + k_{d1} - \frac{\varepsilon \rho}{2} (k_C \bar{q}_d + k_{d2}) \quad (50)$$

$$v_2 := k_{p1} \frac{\varepsilon_2}{1 + \varepsilon_2}. \quad (51)$$

Since the covariance function is continuous and thus, bounded on a closed set  $D$ , the variance  $\Sigma(\tilde{\boldsymbol{\tau}}|\mathbf{p}, \mathcal{D})$  is bounded, for more details see [2]. Thus, there exists

an upper bound  $\bar{\Delta}$  for the model error. It is necessary to ensure that the variables  $v_1, v_2 \in \mathbb{R}_{>0}$  are positive. Therefore, condition (36) must be extended to

$$0 < \varepsilon < \min \left\{ \frac{k_{p1}}{h_2}, \frac{h_1}{h_2}, \frac{2k_{d1}}{2h_2 + \rho(k_C \bar{q}_d + k_{d2})} \right\}. \quad (52)$$

Additionally, they are chosen in a way which keeps the factors of the quadratic parts of the Lyapunov derivative negative. Thus, with (48), equation (46) can be rewritten as (37).  $\square$

Lemmas 3 and 4 provide a valid Lyapunov function and an upper bound for its drift. With these results, Theorem 1 is proven in the following part.

**PROOF ( Theorem 1).** According to [20, Theorem 1] and Lemmas 3 and 4, there exists a  $\xi \in \mathbb{R}_{\geq 0}$  and a  $\varrho \in \mathbb{R}_{\geq 0}$  for (43) such that

$$P \left\{ \dot{V}(\mathbf{x}, t) \leq -\xi V(\mathbf{x}, t) + \varrho \right\} \geq (1 - \delta)^n. \quad (53)$$

Consequently, using [9, Theorem 2.1], the closed loop is uniformly ultimately bounded and exponentially convergent to a ball with a probability of at least  $(1 - \delta)^n$ . Since the state is bounded, it is always possible to find a combination of a set  $D$  and a maximum model error  $\bar{\Delta}$  such that  $\mathbf{p} \in D$ .  $\square$

**Proposition 1** *The radius  $r$  of the ball in (31) is given by*

$$r = \sqrt{\frac{2\varrho}{\xi \min \{k_{p1} - \varepsilon h_2, h_1 - \varepsilon h_2\}}} \quad (54)$$

$$\xi = \frac{2 \min \left\{ \varepsilon v_2, v_1 - \frac{4}{3} \varepsilon k_c \sqrt{\frac{2V_0}{k_{p1} - \varepsilon h_2}} \right\}}{3 \max \{ \varepsilon h_2 + k_{p2}, (1 + \varepsilon) h_2 \}} \quad (55)$$

$$\varrho = \frac{\bar{\Delta}^2}{v_1} + \varepsilon \frac{\bar{\Delta}^2}{v_2} \quad (56)$$

with the extension of (52)

$$0 < \varepsilon < \min \left\{ \frac{k_{p1}}{h_2}, \frac{h_1}{h_2}, \frac{2k_{d1}}{2h_2 + \frac{2k_{p1}\rho^2}{1+\varepsilon_2} + \frac{8}{3}k_c \sqrt{\frac{2V_0}{k_{p1} - \varepsilon h_2}}} \right\}. \quad (57)$$

**PROOF.** See [20]

**Remark 4** *The first summand of (39) contains the influence of the controller on the system while the second summand captures the model error. If a perfect model*

*was available, such that  $\boldsymbol{\mu}(\tilde{\boldsymbol{\tau}}) = \tilde{\boldsymbol{\tau}}(\mathbf{p})$  and thus  $\bar{\Delta} = 0$ , equation (39) with Lemma 2 would show that the closed loop system is asymptotically stable [20].*

**Corollary 1 (Static feedback gains)** *Consider the Lagrangian system (6) and a Gaussian Process trained with (19) which satisfies Assumptions 1 and 2. With Assumption 3 and Lemma 2, the control law (30) with constant feedback matrices  $K_p$  and  $K_d$  guarantees that there exist a closed set  $D$  and a model error  $\bar{\Delta}$  such that*

$$P \left\{ \left\| \begin{matrix} \dot{\mathbf{e}}(t) \\ \mathbf{e}(t) \end{matrix} \right\| \leq r, \forall t \geq t_0 + T(\delta) \right\} \geq (1 - \delta)^n \quad (58)$$

for any initial value  $\|\dot{\mathbf{e}}^\top(t_0), \mathbf{e}^\top(t_0)\| < \delta$  with  $t_0, T(\delta), \delta, r \in \mathbb{R}_{>0}$ .

In the next step, the result will be discussed and examined regarding to its application.

#### 4.1 Design guidelines for feedback gains

Theorem 1 provides an ultimate bound with a given probability depending on the gains, the system parameters and the variance of the GP. The radius of the bound depends quadratically on the upper bound of the model error  $\bar{\Delta}$ . Thus, the radius  $r$  shrinks if the upper bound of the variance of the Gaussian Process Regression decreases. The consequence is an improved tracking performance in terms of tracking error. The posterior variance of the GPR is related to the number and distribution of the training points and can be decreased, for example, with the Bayesian optimization approach where the next training point is set to the position of maximum variance, as proposed in [28]. Especially for the commonly used squared exponential covariance function, each new training point reduces the posterior variance [29].

The lower and upper bound of the adaptive gains also affects the radius of the ball. Since  $\varepsilon$  can be arbitrarily small, an increased lower bound of  $K_d$  significantly shrinks the radius. The influence of  $K_p$  depends directly on the Lagrangian system. Based on the presented results, several design problems can be addressed.

**Design  $K_p, K_d$  such that a ball of predefined radius is achieved:** For this purpose, the model error  $\Delta$  is computed for a desired probability with Lemma 1. Based on the variance of the GPR and thus, on the number and distribution of training points, an upper bound for the model error is determined (49). Afterwards, with (54) the necessary gains can be figured out. **Maximum allowed model variance for a ball of predefined radius:** This allows to draw conclusions regarding the number of required training points. Starting with a desired radius  $r$  and gains  $K_p(\mathbf{q}), K_d(\mathbf{q}, \mathbf{q})$ , (54) is used to determine the maximum model error.



Then, Lemma 1 allows to compute the maximum allowed variance for the GPR on a set  $D$ . The number of training points must be increased until the required maximum variance is achieved.

**Compute the radius of the ball for low feedback gains:** If low feedback gains are predetermined, e.g. for safety reasons, the radius of the ball around the desired trajectory can be computed based on the model error. For the computation of the radius  $r$ , the model error  $\Delta$  with (49) must be computed. Afterwards, (54) shows the resulting radius for the predefined gains  $K_p(\mathbf{q}), K_d(\dot{\mathbf{q}}, \mathbf{q})$ .

## 5 Numerical Illustration

In this section, we present studies illustrating the properties of the proposed CTC-GPR control scheme and a more detailed case study.

### 5.1 Noise attenuation and saturation

In the following example, we show the benefit of the CTC-GPR in comparison to the classical computed torque. For this purpose, we assume a one dimensional Euler-Lagrange-system

$$\tau = \ddot{q} + \dot{q} + q + f_u(\mathbf{p}) \quad (59)$$

with 30 randomly generated dynamics

$$f_u(\mathbf{p}) = \frac{\dot{q}^2 \sin(q - c) - \sin(c)}{\cos(q - c) - 1.1 \cos^{-1}(q - c)} \quad (60)$$

where each  $c$  is uniformly chosen from the set  $[0, 2\pi]$ . For the parametric model, we use the estimates  $\hat{H} = \hat{C} = \hat{g} = 1$ . The 441 training data pairs  $\{\tilde{\tau}\}$  and  $\{\tilde{\mathbf{q}}, \tilde{\dot{\mathbf{q}}}, \tilde{\mathbf{q}}\}$  for a GPR with squared exponential covariance function are equally distributed on the set  $[0] \times [-1, 1] \times [-1, 1]$ . A conjugate gradient algorithm is used to minimize the log likelihood function to find suitable hyperparameters. The desired trajectory is given by  $\mathbf{q}_d = \sin(t)$  and the initial system value is  $q_0 = 0, \dot{q}_0 = 1$ . The measurements of  $\tilde{q}, \tilde{\dot{q}}, \tilde{q}$  are corrupted by Gaussian noise with  $\mathcal{N}(0, 0.04^2)$  for training and control. The simulation time is between zero and  $2\pi$  seconds. In the simulation, the CTC-GPR and the classical computed torque are compared in terms of the maximum tracking error, the noise attenuation and the maximum control action. The feedback gains of the CTC are  $K_p = K_d = 100$  whereas the CTC-GPR is parameterized with

$$K_p(q) = 10 + 100 \Sigma_p(q) \quad (61)$$

$$K_d(\dot{q}, q) = 10 + 100 \Sigma_d(\dot{q}, q). \quad (62)$$

The results are shown in Fig. 3. The variation of the gains is minimal since the desired trajectory is inside the training area where the variance is quite low. The maximal

tracking error  $\max \|\dot{e}(t), e(t)\|$  is decreased compared to CTC approach for all systems with a median of 61.6%. The CTC-GPR shows remarkably better noise attenuation that is denoted by a higher signal to noise ratio (SNR) of the system trajectory. Also the maximal control action is reduced based of the lower feedback gains of the CTC-GPR which can prevent actuator saturation.

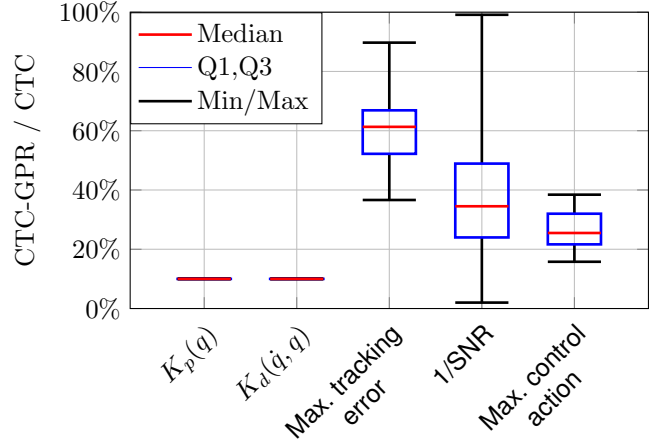


Fig. 3. Comparison between the classical CTC and the proposed CTC-GPR for 30 randomly selected systems. CTC-GPR values are given as a percentage of CTC values.

### 5.2 Case study

In this case study, we apply Lagrange's equations to the model of a 2-link planar manipulator given by [16, Page 164]. We assume point masses for the links of  $m_1 = m_2 = 1$  kg, which are located in the center of each link. The length of the links is set to  $l_1 = l_2 = 1$  m. The joints are without mass and not influenced by any friction. Gravity is assumed to be  $10 \text{ ms}^{-2}$ . As estimates, we use  $\hat{m}_1 = 0.9$  kg,  $\hat{m}_2 = 1.1$  kg,  $\hat{l}_1 = 0.9$  m, and  $\hat{l}_2 = 1.1$  m. Thus, the estimated system matrices of (6) are given by

$$\begin{aligned} \hat{H} &= \begin{bmatrix} 1.41 + 1.09 \cos(q_2) & 0.61 + 0.54 \cos(q_2) \\ 0.61 + 0.54 \cos(q_2) & 0.61 \end{bmatrix} \\ \hat{C} &= \begin{bmatrix} -0.54 \sin(q_2) \dot{q}_2 & -0.54 \sin(q_2) (\dot{q}_1 + \dot{q}_2) \\ 0.54 \sin(q_2) \dot{q}_1 & 0 \end{bmatrix} \\ \hat{g} &= \begin{bmatrix} 9 \sin(q_1) + 6.05 \sin(q_1 + q_2) \\ 6.05 \sin(q_1 + q_2) \end{bmatrix}, \end{aligned} \quad (63)$$

where  $q_1$  and  $q_2$  are the joint angles. The initial values are set to  $\mathbf{q}_0 = [0, 1]^\top$  and  $\dot{\mathbf{q}}_0 = [1, 0]^\top$ . The unknown dynamics  $\mathbf{f}_u(\mathbf{p})$  is simulated by an arbitrarily chosen

nonlinear function

$$\mathbf{f}_u(\mathbf{p}) = \begin{bmatrix} \sin(2\dot{q}_2) + \cos(2q_1) + \ddot{q}_1 \\ \sin(2\dot{q}_2) + 2\sin(\dot{q}_1) \end{bmatrix}. \quad (64)$$

A Gaussian Process with a squared exponential covariance function learns the difference between the estimated model and the true system given by (63) and (64). For this purpose, we generate 576 training pairs on the domain  $\ddot{\mathbf{q}} \in [0, 1]^2$ ,  $\dot{\mathbf{q}} \in [-1, 1]^2$  to generate a set  $\mathcal{D}$  of training points. The measurements of  $\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}$  are corrupted by Gaussian noise with  $\mathcal{N}(0, 0.1^2)$ . The hyperparameters are optimized by means of the likelihood function. The desired trajectory is a sinusoidal function with  $\mathbf{q}_0 = [0, 1]^\top$ . In this example, the gains are adapted according to Lemma 2 with

$$K_p(\Sigma_p(\mathbf{q})) = 7I + 400 \Sigma_p(\mathbf{q}) \quad (65)$$

$$K_d(\Sigma_d(\dot{\mathbf{q}}, \mathbf{q})) = 6I + 400 \Sigma_d(\dot{\mathbf{q}}, \mathbf{q}). \quad (66)$$

Figure 4 shows the resulting trajectory for the first joint along with the desired trajectory (dashed red). As comparison, we use a classic computed torque controller (blue dotted) with the gains  $K_{p,s} = \text{diag}(10, 10)$  and  $K_{d,s} = \text{diag}(10, 10)$ . It becomes apparent that the tracking error of the CTC-GPR approach is lower although the feedback gains are also lower. The color of the trajectory indicates the norm of the current feedback gains for the first joint. In the area close to the training data, the feedback gains remain low (blue color) while outside the training area the gains increase (red color). The result is that the tracking error is kept low and bounded even for areas where no training data is available. A more detailed view of the time-dependent variation of the gains can be seen in Fig. 5. Since the CTC-GPR uses the mean function to compensate the unknown dynamics, the feedback gains can be lower in comparison to the CTC. The advantages of the variable gains are presented in table 1. Here, we also compare the results to a CTC-GPR with static feedback gains. Both CTC-GPR approaches show a lower tracking error than the classic computed torque. Additionally, the variable CTC-GPR outperforms the static CTC-GPR

	CTC	Static CTC-GPR	Variable CTC-GPR
$\ K_p\ $	10	7.02	7.02 - 9.76
$\ K_d\ $	10	6.06	6.06 - 9.44
$\ \mathbf{e}^\top, \dot{\mathbf{e}}^\top\ _{L_2}$	4.7274	1.8771	<b>1.4774</b>
$\max(e_1(t))$	0.2060	0.0877	<b>0.0632</b>
$\max(e_2(t))$	0.1680	0.0614	<b>0.0484</b>
$\max(\dot{e}_1(t))$	0.2345	0.1139	<b>0.0865</b>
$\max(\dot{e}_2(t))$	0.1743	0.0638	<b>0.0548</b>

Table 1  
Comparison between CTC, CTC-GPR with static gains, and CTC-GPR with variable gains.

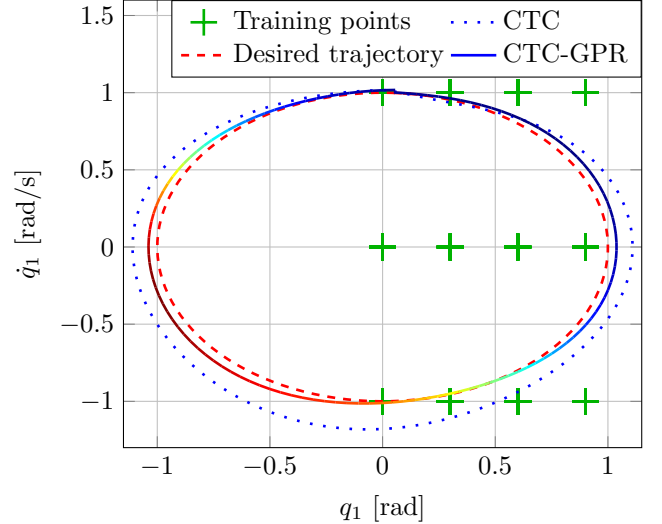


Fig. 4. Tracking performance for the first joint. The color of the CTC-GPR trajectory indicates the norm of the current feedback gains (red high, blue low).

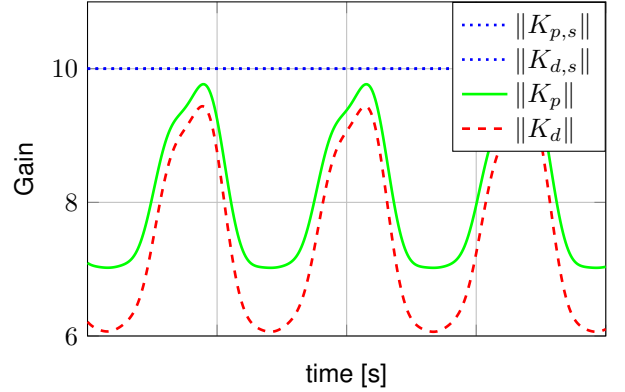


Fig. 5. The norm of the feedback gains for the CTC and the CTC-GPR with variable gains.

for the position and velocity error of both joints because the gains are increased as soon as the trajectory leaves the training area.

## Conclusion

We propose a data-driven approach for high performance tracking control. It is based on a computed-torque control law where the feedback gains are adapted by the model fidelity of a data-driven model of the system. For this purpose, we use Gaussian Process Regression to compensate the residual dynamics of the system. The variance of the GPR is used as a fidelity measure and thus to adapt the feedback gains. The main contribution is that we determine the size of the tracking error of the closed loop system which is proven to be uniformly ultimately bounded and exponentially convergent to a ball with a given probability. The result shows the correlation

between the bound of the tracking error, the uncertainty of the model and the feedback gains. Thus, for specific demands on the closed loop behavior, the result provides the necessary tools for the design of the controller.

## A Proof of Lemma 1

The result is a direct consequence of [26, Theorem 6] which concerns the one dimensional case. In this case, the training data is generated by a scalar function  $f: D \rightarrow \mathbb{R}$  with  $f \in \mathcal{H}_k(D)$  on compact set  $D \subset \mathbb{R}^n$ . A Gaussian Process is trained with  $m$  data points  $\mathcal{D} = \{\mathbf{x}^{\{i\}}, \tilde{y}^{\{i\}}\}_{i=1}^m$  of

$$\tilde{y} = f(\mathbf{x}) + \eta, \quad y, \eta \in \mathbb{R}, \mathbf{x} \in \mathbb{R}^n \quad (\text{A.1})$$

$$\eta \sim \mathcal{N}(0, \sigma), \quad \sigma \in \mathbb{R}_{>0}. \quad (\text{A.2})$$

Then, recalling [26], the model error  $\Delta \in \mathbb{R}$

$$\Delta = |\mu(\mathbf{y}^* | \mathbf{x}^*, \mathcal{D}) - f(\mathbf{x}^*)| \quad (\text{A.3})$$

is bounded with a probability of at least  $(1 - \delta)$  by

$$\mathbb{P} \left\{ \forall \mathbf{x}^* \in D, \Delta \leq |\beta \Sigma^{\frac{1}{2}}(\mathbf{y}^* | \mathbf{x}^*, \mathcal{D})| \right\} \geq (1 - \delta) \quad (\text{A.4})$$

where  $\beta \in \mathbb{R}$  is defined as

$$\beta = \sqrt{2\|f\|_k^2 + 300\gamma \ln^3 \left( \frac{m+1}{\delta} \right)}. \quad (\text{A.5})$$

The variable  $\gamma \in \mathbb{R}$  is the maximum information gain

$$\gamma = \max_{\mathbf{x}^{\{1\}}, \dots, \mathbf{x}^{\{m+1\}} \in D} I(y_j^{\{1\}}, \dots, y_j^{\{m+1\}}; f_j) \quad (\text{A.6})$$

$$= \max_{\mathbf{x}^{\{1\}}, \dots, \mathbf{x}^{\{m+1\}} \in D} \frac{1}{2} \log |I + \sigma_i^{-2} K_{\Phi_j}(\mathbf{x}, \mathbf{x}')| \quad (\text{A.7})$$

with the covariance matrix  $K_{\Phi_j}$  and the used elements  $\mathbf{x}, \mathbf{x}' \in \{\mathbf{x}^{\{1\}}, \dots, \mathbf{x}^{\{m+1\}}\}$ . In the multidimensional case of Lemma 1, we use a GP for each dimension of  $\tilde{\boldsymbol{\tau}}(\mathbf{p})$  as shown in (14). For the calculation of (22), assume the two sets

$$\begin{aligned} \Pi_A &= \left\{ \forall \mathbf{p} \in D, |\mu(\tilde{\tau}_i | \mathbf{p}, \mathcal{D}) - \tilde{\tau}_i(\mathbf{p})| \leq \beta_i \text{var}^{\frac{1}{2}}(\tilde{\tau}_i | \mathbf{p}, \mathcal{D}) \right\} \\ \Pi_B &= \left\{ \forall \mathbf{p} \in D, \|\mu(\tilde{\boldsymbol{\tau}} | \mathbf{p}, \mathcal{D}) - \tilde{\boldsymbol{\tau}}(\mathbf{p})\| \leq \|\beta^\top \Sigma^{\frac{1}{2}}(\tilde{\boldsymbol{\tau}} | \mathbf{p}, \mathcal{D})\| \right\} \end{aligned} \quad (\text{A.8})$$

with the multidimensional extension of  $\gamma$  and  $\beta$

$$\beta_j = \sqrt{2\|\tilde{\tau}_j\|_k^2 + 300\gamma_j \ln^3 \left( \frac{m+1}{\delta} \right)} \quad (\text{A.9})$$

$$\gamma_j = \max_{\mathbf{p}^{\{1\}}, \dots, \mathbf{p}^{\{m+1\}} \in D} \frac{1}{2} \log |I + \sigma_i^{-2} K_{\Phi_j}(\mathbf{x}, \mathbf{x}')| \quad (\text{A.10})$$

$$\mathbf{x}, \mathbf{x}' \in \left\{ \mathbf{p}^{\{1\}}, \dots, \mathbf{p}^{\{m+1\}} \right\}. \quad (\text{A.11})$$

Due to the fact that  $\tilde{\boldsymbol{\tau}}$  is assumed to be uncorrelated (14), the conditional probability for the set  $\Pi_A$  is given by

$$\mathbb{P} \{ \Pi_A \} \geq (1 - \delta)^n. \quad (\text{A.12})$$

With the monotony property of the probability measure  $P$  and since  $\Pi_A \subseteq \Pi_B$  holds, (22) provides an upper bound for the norm of the model error with a probability of at least  $(1 - \delta)^n$ .  $\square$

## B Proof of Lemma 2

According to the Schur's Lemma,  $M$  is negative definite if

$$M_{11} = -K_d(\Sigma_d) + \varepsilon \hat{H} \quad \text{and} \quad (\text{B.1})$$

$$\begin{aligned} S &= -\varepsilon K_p(\Sigma_p) + \frac{\varepsilon^2}{4} (K_d(\Sigma_d) - \hat{C}^\top) \\ &\quad (K_d(\Sigma_d) - \varepsilon \hat{H})^{-1} (K_d^\top(\Sigma_d) - \hat{C}) \end{aligned} \quad (\text{B.2})$$

are negative definite where  $M_{11} \in \mathbb{R}^{n \times n}$  is the upper left block of  $M$  and  $S \in \mathbb{R}^{n \times n}$  is the Schur complement. Since  $K_d$ ,  $\hat{H}$ , and  $K_p$  are positive definite and bounded,  $\varepsilon$  can be chosen sufficiently small to obtain the negative definiteness of  $M_{11}$ . The second summand of the Schur complement  $S$  is quadratic in  $\varepsilon$  and positive definite while the first summand is linear in  $\varepsilon$  and negative. Thus, for every  $\mathbf{q}, \hat{\mathbf{q}} \in \mathbb{R}^n$ , an  $\varepsilon$  can be found which guarantees the negative definiteness of the Schur complement. Therefore, there exists an  $\epsilon > 0$ , so that matrix  $M$  is negative definite.  $\square$

## References

- [1] N.T. Alberto, M. Mistry, and F. Stulp. Computed torque control with variable gains through Gaussian process regression. In *Proc. of the International Conference on Humanoid Robots*, pages 212–217, 2014.
- [2] T. Beckers and S. Hirche. Equilibrium distributions and stability analysis of Gaussian process state space models. In *Proc. of the Conference on Decision and Control*, 2016.
- [3] T. Beckers, J. Umlauf, and S. Hirche. Stable model-based control with Gaussian process regression for robot manipulators. In *Proc. of the 20th IFAC World Congress*, 2017.

- [4] T. Beckers, J. Umlauft, D. Kuli, and S. Hirche. Stable Gaussian process based tracking control of Lagrangian systems. In *Proc. of the Conference on Decision and Control*, 2017.
- [5] F. Berkenkamp, R. Moriconi, A.P. Schoellig, and A. Krause. Safe learning of regions of attraction for uncertain, nonlinear systems with Gaussian processes. In *Proc. of the Conference on Decision and Control*, pages 4661–4666, 2016.
- [6] C.M. Bishop et al. *Pattern recognition and machine learning*, volume 4. Springer New York, 2006.
- [7] C.-A. Cheng, H.-P. Huang, H.-K. Hsu, W.-Z. Lai, and C.-C. Cheng. Learning the inverse dynamics of robotic manipulators in structured reproducing kernel hilbert space. *IEEE Transactions on cybernetics*, 46(7):1691–1703, 2016.
- [8] G. Chowdhary, H.A. Kingravi, J.P. How, and P.A. Vela. Bayesian nonparametric adaptive control using gaussian processes. *IEEE Transactions on neural networks and learning systems*, 26(3):537–550, 2015.
- [9] M. Corless. Guaranteed rates of exponential convergence for uncertain systems. *Journal of Optimization Theory and Applications*, 64(3):481–494, 1990.
- [10] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi. An atlas of physical human–robot interaction. *Mechanism and Machine Theory*, 43(3):253–270, 2008.
- [11] M.P. Deisenroth, D. Fox, and C.E. Rasmussen. Gaussian processes for data-efficient learning in robotics and control. *Transactions on Pattern Analysis and Machine Intelligence*, 37(2):408–423, 2015.
- [12] A. Isidori. *Nonlinear control systems*. Springer Science & Business Media, 2013.
- [13] V. Kapila and K. Grigoriadis. *Actuator saturation control*. CRC Press, 2002.
- [14] K.R. Kozłowski. *Modelling and identification in robotics*. Springer Science & Business Media, 2012.
- [15] K. Kronander and A. Billard. Stability considerations for variable impedance control. *IEEE Transactions on Robotics*, 32(5):1298–1305, 2016.
- [16] R.M. Murray, Z. Li, and S.S. Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 1994.
- [17] D. Nguyen-Tuong, M. Seeger, and J. Peters. Computed torque control with nonparametric regression models. In *American Control Conference, 2008*, pages 212–217. IEEE, 2008.
- [18] R. Ortega and M.W. Spong. Adaptive motion control of rigid robots: A tutorial. *Automatica*, 25(6):877–888, 1989.
- [19] C.E. Rasmussen. *Gaussian processes for machine learning*. MIT Press, 2006.
- [20] T. Ravichandran, D.W.L. Wang, and G.R. Heppler. Stability and robustness of a class of nonlinear controllers for robot manipulators. In *Proc. of the American Control Conference*, volume 6, pages 5262–5267. IEEE, 2004.
- [21] L. Sciavicco and B. Siciliano. *Modelling and control of robot manipulators*. Springer Science & Business Media, 2012.
- [22] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: modelling, planning and control*. Springer Science+Business Media, 2010.
- [23] J.-J.E. Slotine and W. Li. On the adaptive control of robot manipulators. *The international journal of robotics research*, 6(3):49–59, 1987.
- [24] Mark W Spong and Mathukumalli Vidyasagar. *Robot dynamics and control*. John Wiley & Sons, 2008.
- [25] M.W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot modeling and control*, volume 3. wiley New York, 2006.
- [26] N. Srinivas, A. Krause, S.M. Kakade, and M.W. Seeger. Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012.
- [27] I. Steinwart and A. Christmann. *Support vector machines*. Springer Science & Business Media, 2008.
- [28] Y. Sui, A. Gotovos, J. Burdick, and A. Krause. Safe exploration for optimization with gaussian processes. In *Proc. of the International Conference on Machine Learning*, pages 997–1005, 2015.
- [29] J. Umlauft, T. Beckers, M. Kimmel, and S. Hirche. Feedback linearization using Gaussian processes. In *Proc. of the Conference on Decision and Control*. IEEE, 2017.
- [30] J. Vinogradska, B. Bischoff, D. Nguyen-Tuong, A. Romer, H. Schmidt, and J. Peters. Stability of controllers for gaussian process forward models. In *International Conference on Machine Learning*, pages 545–554, 2016.
- [31] G. Wahba. *Spline models for observational data*. SIAM, 1990.