Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Constrained Gaussian Process Learning for Model Predictive Control ⋆

Janine Matschek * **Andreas Himmel** ** **Kai Sundmacher** **,*** **Rolf Findeisen** *

* *Laboratory for Systems Theory and Automatic Control, Otto von Guericke University Magdeburg, Germany (e-mail: {janine.matschek, rolf.findeisen}@ovgu.de)*
** *Process Systems Engineering, Otto von Guericke University Magdeburg, Germany*
*** *Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany*

Abstract: Many control tasks can be formulated as tracking problems of a known or unknown reference signal. Examples are motion compensation in collaborative robotics, the synchronisation of oscillations for power systems or the reference tracking of recipes in chemical process operation. Both the tracking performance and the stability of the closed-loop system depend strongly on two factors: Firstly, they depend on whether the future reference signal required for tracking is known, and secondly, whether the system can track the reference at all. This paper shows how to use machine learning, i.e. Gaussian processes, to learn a reference from (noisy) data while guaranteeing trackability of the modified desired reference predictions within the framework of model predictive control. Guarantees are provided by adjusting the hyperparameters via a constrained optimisation. Two specific scenarios, i.e. asymptotically constant and periodic references, are discussed.

Keywords: Gaussian processes, trajectory tracking, learning supported model predictive control

1. INTRODUCTION

Model predictive control (MPC) is a popular optimisation based control strategy which can handle a broad class of dynamical systems including nonlinear, constrained, multi-input multi-output systems. MPC can be used for different control tasks, including setpoint stabilisation, tracking of time dependent references, path following or economical operation of a system, see for example (Matschek et al., 2019). To guarantee repeated feasibility of the optimal control problem as well as to achieve closed loop stability several concepts exist that differ depending on the control tasks. In tracking MPC the controller can e.g. be designed in error coordinates which leads to time-varying error dynamics and consequently time-dependent terminal ingredients to prove stability (Faulwasser and Findeisen, 2011). To do so, the reference needs to be known and trackable for the system, i.e. it must be compliant with the state constraints and an admissible reference input should exist to follow the reference given the system dynamics. Alternatively, one can use artificial references (Limon et al., 2008, 2012; Ferramosca et al., 2009) to ensure feasibility under changing references. Hereby, the system state is steered to follow an artificial reference while the distance of the artificial reference to the actual reference is minimized. Reference modification to achieve good performance and stability can also be achieved by reference governors, which act as pre-filters for the reference signal, see e.g. (Garone et al., 2017) and references therein.

Machine learning, as for instance a Gaussian process (GP), can be used to obtain analytical models of external reference signals that are only available in terms of (noisy) data. Application examples are dynamically operated chemical plants where references are obtained via real-time optimisation or autonomous cars which learn from and adapt to human driver provided references. Learning of external or additive signals via GPs is considered for instance in (Maiworm et al., 2018; Klenske et al., 2016; Matschek et al., 2020), where feed-forward control signals and external reference signals are learned, respectively. In this work, additional constraints in the learning of GPs are included to guarantee that the signal provided by the GP is suitable for a predictive controller. In other words, the learned reference should be trackable or at least guarantee recursive feasibility. This is important as noise in a reference can lead to infeasibility of the controller even if the underlying true signal is trackable. Moreover, adding constraints can improve the approximation quality of the GP as unrealistic evolutions are excluded. In case that the original reference is not trackable, the constrained GP allows to find a trade-off between close approximation and constraint satisfaction. Thus the GP is used for reference prediction and adaptation if necessary, cf. Figure 1.

The main contributions of this paper is a guideline how to setup and train GPs to be used as reference predictors with guaranteed trackability of the learned reference.

⋆ The work of this paper is supported by the Federal Ministry of Education and Research within the Forschungscampus *STIMULATE* under grant number '13GW0095A'.

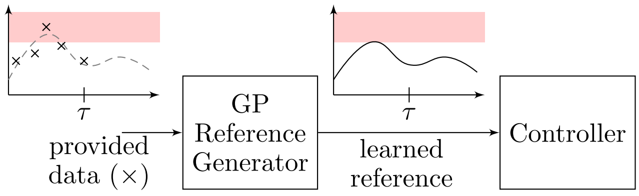

Fig. 1. Based on data, denoted by $\times$, the unknown reference (dashed) should be modelled by the GP. It learns a reference (solid) while both prediction into the future ($t > \tau$) as well as reference shaping to satisfy the constraints (red) is achieved.

These learning algorithms utilize special structures of the underlying reference which should be modelled, as well as constrained hyperparameter estimation. In contrast to (Da Veiga and Marrel, 2012) we do not use truncated multinormal distributions, but constrain the predicted mean of the GP to lie inside the reachable tube and the state constraints during hyperparameter estimation.

The remainder of this paper is structured as follows: Section 2 describes the problem setup of learning based reference prediction via Gaussian processes. Section 3 proposes algorithms for Gaussian processes training to guarantee trackability of the reference. Section 4 summarises the achievements and provides directions for future work.

2. PROBLEM SETUP

Consider the nonlinear time-discrete system

$$x(k+1) = f(x(k), u(k)), \quad x(0) = x_0, \tag{1}$$

where $x \in \mathbb{R}^{n_x}$ is the state, $u \in \mathbb{R}^{n_u}$ is the input, and $x_0 \in \mathbb{R}^{n_x}$ is the initial condition of the system.

In tracking MPC, the goal is to design a controller such that the system state follows a reference x_r while satisfying state constraints $\mathcal{X}$ and input constraints $\mathcal{U}$. One possibility to guarantee stability of tracking MPC is the use of time varying terminal equality or inequality constraints which depend on the reference (Faulwasser and Findeisen, 2011; Rawlings et al., 2017; Matschek et al., 2019). Determination of these terminal ingredients consequently requires the knowledge of the reference x_r, which is however not always a priori known. For example, dynamic operation of chemical plants might lead to sudden changes in the reference based on economical considerations. Other examples are autonomous vehicles following a human driver (e.g. adapting to its velocity) while not knowing its future decisions. In such cases, machine learning can be used to obtain a model of the reference, such that the reference can be predicted and thus is known to the controller including predictions of future values. Besides the required knowledge of the reference, the reference must fulfil the following property:

Definition 1. (Constrained Trackability).
A reference $x_\mathrm{r} : \mathbb{N}_0 \to \mathbb{R}^{n_x}$ is said to be trackable for system (1) if it fulfils the state constraints $x_\mathrm{r}(k) \in \mathcal{X}$ and can be followed given the system dynamics, i.e. $\exists u_\mathrm{r}(k) \in \mathcal{U}$ such that $x_\mathrm{r}(k+1) = f(x_\mathrm{r}(k), u_\mathrm{r}(k))$ for all $k \in \mathbb{N}_0$.

Though restrictive, trackability of the reference according to Definition 1 enables desirable properties such as recursive feasibility of an MPC with terminal equality constraints (once being on the reference there exists an admissible input to stay on it). This definition allows us to formulate the following task:

Task 1. (Reference generator).
Given system (1) and data/ measurements $\mathcal{D} := \prod_{i=0}^{n} \mathbb{R}_0^+ \times \mathbb{R}^{n_x}$ describing the desired reference $r : \mathbb{N}_0 \to \mathbb{R}^{n_x}$. Design a reference generator $g : \mathbb{N}_0 \times \mathcal{D} \to \mathcal{X}$, which provides for $D \in \mathcal{D}$ a reference $x_\mathrm{r} : \mathbb{N}_0 \to \mathcal{X}, (k) \mapsto x_\mathrm{r}(k) := g(k, D)$, which fulfils:

(i) Trackability: The reference x_r is trackable.
(ii) Reference Prediction: The reference x_r spans at least over a receding prediction horizon N, i.e. at $k, x_\mathrm{r}(i)$ is known $\forall i \in \{k, k+1, \ldots, k+N\}$.
(iii) Data fitting: The reference model finds a trade off between model complexity and data consistency, i.e. $x_\mathrm{r}(k) \approx r(k)$ for $D \in \mathcal{D}$.

To address this task we propose to use a machine learning technique called Gaussian processes. A general introduction to GPs and an elaboration on the use of them as reference generators is provided in the following.

2.1 Gaussian processes

Gaussian Processes are stochastic modelling approaches which can be used for classification and regression problems. In control they have gained an increasing attention for the modelling of both static and dynamic systems, see e.g. (Rasmussen and Williams, 2006; Ostafew et al., 2016; Kocijan et al., 2004; Berkenkamp and Schoellig, 2015). Reasons for this popularity are the limited amount of design decisions, their capability of dealing with noisy data, and the confidence interval that is provided by the GP which allows to investigate the quality of the model.

We use GPs to obtain the desired reference generator. Here, the GP can be uniquely defined by a mean function $m : \mathbb{R} \to \mathbb{R}$ and a symmetric, positive semi-definite covariance function $\kappa : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_0^+$ and is denoted by

$$y(t) \sim \mathcal{GP}(m(t), \kappa(t, t')).$$

Here, $t, t' \in \mathbb{R}$ are the regressors or inputs to the GP and the values of the process $y(t)$ at each specific time t possess a normal distribution. Via the covariance function (also called kernel) a GP relates similarities between the input variables to the similarity between the output variables. These mean and covariance functions involve hyperparameters $\theta \in \mathbb{R}^{n_\theta}$, where n_θ depends on the selected functions m and κ. The values of the hyperparameters can be learned based on a hyperparameter training set $D_\theta := \left\{ (t_{\theta,i}, y_{\theta,i}) \in \mathbb{R}_0^+ \times \mathbb{R}^{n_x} \,|\, i = 1, 2, \ldots, n_{D_\theta} \right\} \in \mathcal{D}$. To do so, often a point estimate of θ is calculated via the maximization of the marginal logarithmic likelihood.

Our overall goal is to predict or infer the distribution of the output at (possibly unseen) test points t_*. This prediction is based on several design decisions, the hyperparameters, and a training data set $D_\mathrm{t} := \left\{ (t_{\mathrm{t},i}, y_{\mathrm{t},i}) \in \mathbb{R}_0^+ \times \mathbb{R}^{n_x} \,|\, i = 1, 2, \ldots, n_{D_\mathrm{t}} \right\} \in \mathcal{D}$. For ease of notation, we define

$$\boldsymbol{t} := [t_{\mathrm{t},1}, \ldots, t_{\mathrm{t},n_{D_\mathrm{t}}}], \tag{2a}$$

$$\boldsymbol{y} := [y_{\mathrm{t},1}, \ldots, y_{\mathrm{t},n_{D_\mathrm{t}}}], \tag{2b}$$

$$\boldsymbol{m}(\boldsymbol{t}) := [m(t_{\mathrm{t},1}), \ldots, m(t_{\mathrm{t},n_{D_\mathrm{t}}})] \tag{2c}$$

The joint distribution of the training data output $\boldsymbol{y}$ and the test data output y_* at t_* can be expressed as

$$\begin{pmatrix} \boldsymbol{y} \\ y_* \end{pmatrix} \sim \mathcal{GP}\left(\begin{pmatrix} \boldsymbol{m}(\boldsymbol{t}) \\ m(t_*) \end{pmatrix}, \begin{pmatrix} K(\boldsymbol{t},\boldsymbol{t}) + \sigma_n^2 I & K(\boldsymbol{t}, t_*) \\ K(t_*, \boldsymbol{t}) & \kappa(t_*, t_*) \end{pmatrix} \right).$$

Here, σ_n^2 represents the variance of the measurement noise. The entries of the covariance matrix K are calculated using the covariance function κ. Specifically, $K(\boldsymbol{t},\boldsymbol{t})$ is of dimension $n_{D_\mathrm{t}} \times n_{D_\mathrm{t}}$ and specifies the covariance between all of the training data points, while $K(\boldsymbol{t}, t_*)$ and $K(t_*, \boldsymbol{t})$ (with dimensions $n_{D_\mathrm{t}} \times 1$ and $1 \times n_{D_\mathrm{t}}$, respectively) define the cross correlation between test and training data points. The scalar $\kappa(t_*, t_*)$ is the auto covariance of the test data. Given this joint probability distribution, the conditional posterior distribution can be calculated via the posterior mean function $m^+ : \mathbb{R} \to \mathbb{R}$ defined by

$$\begin{aligned} m^+(t_*) :=& m(t_*) \\ &+ K(t_*, \boldsymbol{t})(K(\boldsymbol{t},\boldsymbol{t}) + \sigma_n^2 I)^{-1}(\boldsymbol{y} - \boldsymbol{m}(\boldsymbol{t})) \end{aligned} \tag{3}$$

and the posterior covariance $\kappa^+ : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_0^+$ defined by

$$\begin{aligned} \kappa^+(t_*, t_*) :=& \kappa(t_*, t_*) \\ &- K(t_*, \boldsymbol{t})\big(K(\boldsymbol{t},\boldsymbol{t}) + \sigma_n^2 I\big)^{-1} K(\boldsymbol{t}, t_*). \end{aligned}$$

These posterior moments clearly depend on the involved data set $D = D_\theta \cup D_\mathrm{t}$ and the hyperparameter θ. Therefore, whenever necessary we will explicitly denote this dependency by $m^+(t_*|D, \theta)$ and $\kappa^+(t_*, t_*|D, \theta)$, but refrain from using it elsewhere for sake of brevity of notation.

2.2 GPs as reference predictors

In our setup we will use the posterior mean of the GP as the reference $x_\mathrm{r}(k) := m^+(t_*)$, where $t_* = T_\mathrm{s}k \in \mathfrak{T} := \{t \in \mathbb{R}_0^+ | \, t = T_\mathrm{s}k, k \in \mathbb{N}_0\}$ with sampling time T_s. If $n_x > 1$, either n_x independent GPs can be trained as e.g. done in (Matschek et al., 2020) or correlations between the outputs can be modelled, see e.g. (Rasmussen and Williams, 2006; Salzmann and Urtasun, 2010).

By using Gaussian processes as reference predictors Task 1 (ii) and (iii) are naturally fulfilled, as GPs form a (static) prediction model trained via a hyperparameter optimisation that avoids overfitting. The remaining task is to satisfy the trackability property. The reference x_r and therefore the posterior mean $m^+(t_*)$ with $t_* \in \mathfrak{T}$ must be consistent with state constraints. Additionally, it must be followable for system (1) such that there exists an input $u(k) \in \mathcal{U}$, $\forall k \in \mathbb{N}_0$ for system (1) to stay on the reference when starting on it. To this end we use the definition of a reachable set from Blanchini and Miani (2008):

Definition 2. (Reachability set). Given the set of initial conditions $\mathcal{P} \subset \mathbb{R}^{n_x}$, the reachability set $\mathcal{R}_T(\mathcal{P}) \subset \mathbb{R}^{n_x}$ from $\mathcal{P}$ in time $T < +\infty$ is the set of all states x for which there exists $x(0) \in \mathcal{P}$ and $u(\cdot) \in \mathcal{U}$ such that $x(T) = x$.

We can use the one step ahead reachable set $\mathcal{R}_1(\mathcal{P})$ from an initial condition $\mathcal{P} := \{x_\mathrm{r}(k)\}$ as a sufficient condition to verify that $x_\mathrm{r}(k+1)$ is trackable, i.e. if $x_\mathrm{r}(k+1) \in \mathcal{R}_1\big(x_\mathrm{r}(k)\big)$ and $x_\mathrm{r}(k) \in \mathcal{X}$ for all $k \in \mathbb{N}_0$ then the reference is trackable according to Definition 1. We denote the one step ahead reachable tube as $\mathcal{T}_{k+1} := \mathcal{R}_1(x_\mathrm{r}(k))$. An illustration of a reachable tube is shown in Figure 2. In essence, we have to design the GP such that the posterior mean $m^+(t_* = T_\mathrm{s}k)$ is constrained by the intersection of the state constraints $\mathcal{X}$ and the reachable tube $\mathcal{T}_k$. Then, point three of Task 1 is fulfilled. We show how to do so in the following paragraphs.

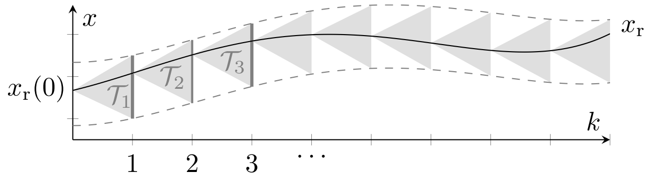

Fig. 2. Illustration of a reachable tube.

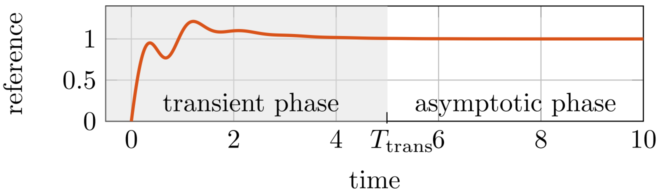

Fig. 3. Illustration of an asymptotically constant reference.

3. CONSTRAINED GP LEARNING

We propose to include constraints in the learning phase of the GP to satisfy $m^+(t_*) \in (\mathcal{T}_k \cap \mathcal{X})$ with $t_* = T_\mathrm{s}k$ by this ensuring trackability. To learn the hyperparameters θ of the mean and covariance function we rely on maximising the logarithmic marginal likelihood. This optimisation results in a point estimate of the most likely hyperparameters given the prior belief (uniformly distributed hyperparameter prior) and the hyperparameter training data set $D_\theta \in \mathcal{D}$. Analogously to (2) we define $\boldsymbol{t}_\theta, \boldsymbol{y}_\theta$ and $\boldsymbol{m}(\boldsymbol{t}_\theta)$ from D_θ. The optimisation problem can be written as

$$\hat{\theta} := \underset{\theta}{\arg\min} \quad l(\theta) \tag{4a}$$

$$\text{subject to} \quad m^+(t_*|D_\theta, \theta) \in (\mathcal{T}_k \cap \mathcal{X}), \tag{4b}$$

$$t_* = T_\mathrm{s}k, \ \forall k \in \{0, \ldots, \bar{k}\}, \tag{4c}$$

where the cost function $l(\theta)$ is the negative logarithmic marginal likelihood

$$l(\theta) := \ln(|K(\boldsymbol{t}_\theta, \boldsymbol{t}_\theta)|) + \boldsymbol{y}_\theta^\top K(\boldsymbol{t}_\theta, \boldsymbol{t}_\theta)^{-1}\boldsymbol{y}_\theta + n_{D_\theta} \ln(2\pi).$$

We denote the optimal solution of (4) with $\hat{\theta}$. Despite the small number of decision variables θ, the optimisation problem is complex as it has a large number of constraints, is nonconvex, and requires the inverse of $K(\boldsymbol{t}_\theta, \boldsymbol{t}_\theta)$ which can be large for a high number of data points n_{D_θ}. To solve the optimisation problem several numerical optimisation methods exist, see e.g. Kocijan (2016). In the remainder of this paper, we rely on the following assumption:

Assumption 1. The optimisation problem (4) is feasible.

Proposition 1. Given Assumption 1 the resulting parametrisation $\hat{\theta}$ of the GP obtained via problem (4) guarantees trackability of the reference for system (1) for all $k \in \{0, 1, \ldots, \bar{k}\}$.

Proposition 1 only guarantees trackability up to step $\bar{k}$. To be able to guarantee trackability for all times, as demanded in Task 1, we consider asymptotic as well as periodic references as special cases in the following.

3.1 Asymptotically Constant References

A special case of time dependent references are those which change for a finite time and are constant (or converge to a constant) afterwards, cf. Figure 3. Examples for such references are the transition between two setpoints in chemical plants or the parking of a car. As the transient phase is of finite time $T_{\text{trans}} = T_{\text{s}} k_{\text{trans}} < \infty$, a finite number of constraints allows for trackability during the transient via $\bar{k} \geq k_{\text{trans}}$. In the following, an iterative algorithm is derived which ensures trackability for $k \geq \bar{k}$. For the specific type of reference we use a prior mean m and covariance function κ with following properties:

Assumption 2. The prior mean function m is constant.

Assumption 3. The covariance function κ is stationary and strictly monotonously decreasing $\kappa(t_1, t_2) < \kappa(t_3, t_4)$ for all $|t_2 - t_1| > |t_4 - t_3|$.

Assumption 4. The absolute value of the time derivative of the covariance function $\dot{\kappa} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}, \dot{\kappa} := \frac{\partial \kappa(\cdot, t')}{\partial t}$ is strictly monotonously decreasing $|\dot{\kappa}(t_1, t_2)| < |\dot{\kappa}(t_3, t_4)|$ at least for all $|t_2 - t_1| > |t_4 - t_3| > \zeta(\theta)$, where $\zeta : \mathbb{R}^{n_\theta} \to \mathbb{R}$ depends on the hyperparameters.

For example, the popular squared exponential covariance function $\kappa(t, t') = \theta_1^2 \exp(-\frac{1}{2\theta_2^2}(t - t')^2)$ fulfils Assumption 3 and 4 for $\zeta(\theta) = \theta_2$. Assuming fixed hyperparameters θ and a fixed training data set $D_\theta \in \mathcal{D}$, the posterior mean (3) can be reformulated as a weighted sum

$$m^+(t_*) = m(t_*) + \sum_{i=1}^{n_{D_\theta}} c_i \kappa(t_i, t_*). \tag{5}$$

Here, $t_i \in \boldsymbol{t}_\theta$ from D_θ and c_i are constant coefficients which depend on the fixed hyperparameters and the training data. A bound on m^+ can be obtained via

$$|m^+(t_*) - m(t_*)| \leq \bar{m}(t_*) := \sum_{i=1}^{n_{D_\theta}} |c_i| \kappa(t_i, t_*). \tag{6}$$

Similar to (5) and (6) and with Assumption 2, the derivative of m^+ can be expressed via

$$\dot{m}^+(t_*) = \sum_{i=1}^{n_{D_\theta}} c_i \dot{\kappa}(t_i, t_*),$$

with a corresponding bound

$$|\dot{m}^+(t_*)| \leq \bar{\dot{m}}(t_*) := \sum_{i=1}^{n_{D_\theta}} |c_i| |\dot{\kappa}(t_i, t_*)|. \tag{7}$$

Furthermore, we rely on the assumption that the growth rate of the tube can be characterised by a constant lower bound $\underline{\tau}$ and upper bound $\bar{\tau}$:

Assumption 5. $\underline{\tau}$ and $\bar{\tau}$ are constant and form an inner approximation of the one-step reachable tube such that $[x_{\text{r}}(k) - \underline{\tau} T_{\text{s}}, x_{\text{r}}(k) + \bar{\tau} T_{\text{s}}] \subseteq \mathcal{T}_{k+1}$ for all $x_{\text{r}}(k) \in \tilde{\mathcal{X}} \subseteq \mathcal{X}$.

Let $\tilde{\mathcal{X}} := [m(\bar{t}) - \bar{m}(\bar{t}), m(\bar{t}) + \bar{m}(\bar{t})]$. If $\tilde{\mathcal{X}} \subseteq \mathcal{X}$, and if $\underline{\tau} \leq \bar{\dot{m}}(\bar{t}) \leq \bar{\tau}$ for time $\bar{t} = \bar{k} T_{\text{s}}$, then (relying also on (4)) $m(t) \in \mathcal{X} \cap \mathcal{T}_k(\mathcal{P})$ for all $t \in \mathfrak{T}$, i.e. trackability is achieved as also outlined in Lemma 1. If these requirements are not fulfilled for $\bar{k}$, the hyperparameter optimisation (4) must be performed again with updated $\bar{k}$. Iteratively updating the number of constraints by increasing $\bar{k}$ will lead to constraint satisfaction for a longer time span, decreased bounds for the mean and its derivative (as $|t_i - \bar{t}|$ is increased) as well as less conservative bounds $\underline{\tau}$, $\bar{\tau}$. In Algorithm 1, the whole procedure is summarised.

We assume that Algorithm 1 terminates in finite time. This allows to conclude the following result:

Algorithm 1 GP Learning for asymptotic references

1: **procedure** GP Training
2:     **Init** $\bar{k}, \mathcal{D}_\theta$
3:     **while** true **do**
4:         obtain $\hat{\theta}$ via (4) using $\left(\bar{k}, \mathcal{D}_\theta\right)$
5:         **if** $|t_i - T_{\text{s}} \bar{k}| > \zeta\left(\hat{\theta}\right)$ **then**
6:             compute $m^+(\bar{t}), \bar{m}(\bar{t}), \bar{\dot{m}}(\bar{t})$ with $\bar{t} = T_{\text{s}} \bar{k}$ via (3),(6),(7) using $\left(\hat{\theta}, \mathcal{D}_\theta\right)$
7:             **if** $m^+(\bar{t}) \pm \bar{m}(\bar{t}) \in \mathcal{X}$ **then**
8:                 choose $\underline{\tau}, \bar{\tau}$ in accordance to Ass. 5 with $\tilde{\mathcal{X}} = [m^+(\bar{t}) - \bar{m}(\bar{t}), m^+(\bar{t}) + \bar{m}(\bar{t})]$
9:                 **if** $\underline{\tau} \leq \bar{\dot{m}}(\bar{t}) \leq \bar{\tau}$ **then**
10:                     **break**
11:         $\bar{k} \leftarrow \bar{k} + 1$
12:     **return** $\bar{k}, \hat{\theta}$

Lemma 1. Given Assumptions 1 to 5 the posterior mean (3) of a GP trained with Algorithm 1 is trackable in the sense of Definition 1 for system (1).

Proof 1. If Algorithm 1 converges, we obtain the optimised hyperparameters $\hat{\theta}$ and the time instant $\bar{k}$. Problem (4) (under Assumption 1) guarantees $m^+(t) \in (\mathcal{T}_k \cap \mathcal{X})$ for all $t \in \mathfrak{T}_{\leq} := \{t \in \mathfrak{T} | t \leq \bar{t} = \bar{k} T_{\text{s}}\}$. From Assumption 3 follows $\bar{m}(t) < \bar{m}(\bar{t})$ if $|t_i - t| > |t_i - \bar{t}|$ for all $t_i \in \boldsymbol{t}_\theta$. Line 7 in Algorithm 1 guarantees $[m(\bar{t}) - \bar{m}(\bar{t}), m(\bar{t}) + \bar{m}(\bar{t})] \in \mathcal{X}$ and thus $m^+(t) \in \mathcal{X}$ for all times $t \in \mathfrak{T}_{>} := \mathfrak{T} \setminus \mathfrak{T}_{\leq}$. Line 5 in Algorithm 1 ensures monotonicity of $|\dot{\kappa}|$ (see Assumption 4). Additionally, $\underline{\tau} \leq \bar{\dot{m}}(\bar{t}) \leq \bar{\tau}$ (line 9). Due to the structure of (7) monotonicity of $|\dot{\kappa}|$ implies monotonicity of $\bar{\dot{m}}$ such that $\bar{\dot{m}}(t) < \bar{\dot{m}}(\bar{t})$ for all t which fulfil $|t_i - t| > |t_i - \bar{t}|$, with $t_i \in \boldsymbol{t}_\theta$. Consequently, $\underline{\tau} \leq \bar{\dot{m}}(t) \leq \bar{\tau}$ for all $t \in \mathfrak{T}_{>}$. Including Assumption 5 (line 8) results in $m^+(t + T_{\text{s}}) \in [m^+(t) - \underline{\tau} T_{\text{s}}, m^+(t) + \bar{\tau} T_{\text{s}}] \subseteq \mathcal{T}_{k+1} \, \forall t \in \mathfrak{T}_{>}$. In other words, $m^+(t) \in \mathcal{T}_k$ for $t \in \mathfrak{T}_{>}$. All in all, $m^+(t) \in (\mathcal{T}_k \cap \mathcal{X})$ for $t \in \mathfrak{T}$. □

We provide an illustrative example in the following:

Example 1. Given is a dynamical system $x(k+1) = 0.9x(k) + 0.5u(k)$ with state constraints $\mathcal{X} = [-2, 0.05]$ and input constraints $\mathcal{U} = [-0.5, 0.5]$. We want to model the reference depicted by the black solid line in Fig. 5, which is however unknown to the GP. Only data points (depicted as crosses), which cover the transient phase and the fact that the reference converges to a constant after the transient are known. We choose a constant zero prior mean function $m(t_*) = 0$ and the squared exponential covariance function $\kappa(t_i, t_*) = \theta_1^2 \exp(-\frac{1}{2\theta_2^2}(t_* - t_i)^2)$. Algorithm 1 is initialised with $\bar{k} = 70 > k_{\text{trans}}$. The convergence of the bounds $\bar{m}, \bar{\dot{m}}$ as well as the optimal hyperparameters $\hat{\theta}$ at each iteration are depicted in Figure 4. Monotonicity of $|\dot{\kappa}|$ (line 5) is fulfilled after $k = 81$ only. The convergence of $\bar{m}$ and $\bar{\dot{m}}$ is mainly influenced by the increase of the distance $|t_i - t_*|$. After $\bar{k} = 101$ the inner tube approximation is non empty (line 9 is feasible) and after $\bar{k} = 108$ the bound on the derivative lies inside of it (line 9 is fulfilled). The iterative algorithm is terminated at $\bar{k} = 130$ as the bound for the predicted mean becomes small enough ($\bar{m} = 0.0468$) to guarantee trackability for all times. The resulting GP prediction is depicted in Figure 5 in blue dashed line. In contrast to a GP whose hyperparameters were conventionally optimised without

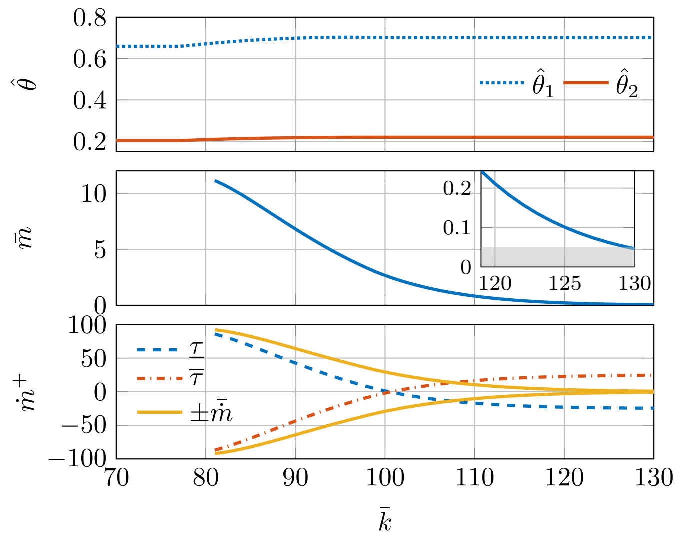

Fig. 4. Optimal hyperparameters $\hat{\theta}$, the bound for the mean value $\bar{m}$, and the bound on its derivative $\bar{\dot{m}}$ compared to the inner approximation of the tube growth $\overline{\tau}, \underline{\tau}$ at each iteration of Algorithm 1 for Example 1. The zoom shows the last 11 iterations.

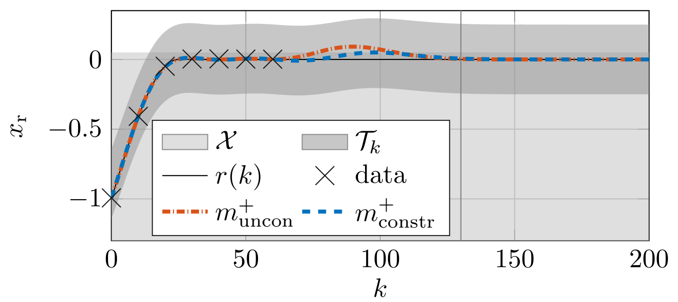

Fig. 5. Comparison between an unconstrained GP (dash-dotted) and the proposed learning algorithm (dashed). The constraints for the hyperparameter optimisation $\mathcal{X} \cap \mathcal{T}_k$ (dark grey), are used up to $\bar{k} = 130$.

additional constraints (depicted in red dash-dotted line) it satisfies trackability for all times.

In systems where not every state is directly influenced by an input, the lower and upper bound $\underline{\tau}$ and $\overline{\tau}$ become equal for some states. In this case line 9 of Algorithm 1 cannot be fulfilled. Instead of using the one step reachable set, the N step reachable set can then be used. This relaxes the restriction significantly and guarantees recursive feasibility instead of trackability.

3.2 Periodical references

Often periodic references are of interest. They occur e.g. in the periodic operation of chemical reactors or when a robot manufactures the same item iteratively. To be able to extrapolate periodic signals with Gaussian processes we rely on Assumption 2 and the following covariance:

Assumption 6. Covariance κ is stationary and periodic, i.e. $\kappa(t, t') = \kappa(t, t' + nT_\text{p})$ with $n \in \mathbb{N}_0$ and period T_p.

To guarantee trackability, the constraints of the optimisation (4) should cover at least one period $\bar{k} \geq T_\text{p}/T_\text{s}$. If the period T_p is an integer multiple of the sampling time T_s then (4) with $\bar{k} \geq T_\text{p}/T_\text{s}$ guarantees trackability for all times $t \in \mathfrak{T}$. In the more general case, constraint satisfaction must be guaranteed not only point wise but ultimately for all $t \in [0, T_\text{p}]$. Therefore, we will derive bounds on the mean and its derivative to guarantee constraint satisfaction not only at the sampling instances $t \in \mathfrak{T}$, but for all $t \in [0, T_\text{p}]$. To this end, we propose the optimisation:

$$\hat{\theta} := \arg\min_{\theta} l(\theta) \tag{8a}$$

$$\text{s. t.}$$

$$m^+_{\min} = \min_t m^+(t|D_\theta, \theta), \quad \text{s.t. } 0 \leq t \leq T_\text{s}\bar{k}, \tag{8b}$$

$$m^+_{\max} = \max_t m^+(t|D_\theta, \theta), \quad \text{s.t. } 0 \leq t \leq T_\text{s}\bar{k}, \tag{8c}$$

$$\dot{m}^+_{\min,i} = \min_t \dot{m}^+(t|D_\theta, \theta), \quad \text{s.t. } \Delta t_i \leq t \leq \Delta t_{i+1}, \tag{8d}$$

$$\dot{m}^+_{\max,i} = \max_t \dot{m}^+(t|D_\theta, \theta), \quad \text{s.t. } \Delta t_i \leq t \leq \Delta t_{i+1}, \tag{8e}$$

$$m^+_{\min} \in \mathcal{X}, \quad m^+_{\max} \in \mathcal{X}, \tag{8f}$$

$$\dot{m}^+_{\min,i} \geq \underline{\tau}_i, \quad \dot{m}^+_{\max,i} \leq \overline{\tau}_i, \tag{8g}$$

$$\forall i \in [0, 1, \ldots, \eta - 1], \qquad \Delta t_i := i\eta^{-1}T_\text{s}\bar{k}. \tag{8h}$$

Here, a constant upper and lower bound on the mean via (8b) and (8c) is obtained which need to fulfil the state constraints (8f). As a constant bound for the derivative of the mean and the inner tube approximation could be quite conservative, piecewise constant approximations are used. The time span $[0, T_\text{s}\bar{k}]$ is therefor divided into η intervals. For each interval, a lower bound $\dot{m}^+_{\min,i}$ and upper bound $\dot{m}^+_{\min,i}$ on the derivative of the mean is calculated in (8d) and (8e). Furthermore, inner approximations of the reachable tube growth $[\underline{\tau}_i, \overline{\tau}_i]$ for $\tilde{\mathcal{X}}_i := \{m^+(t)\, \forall t \in [\Delta t_i, \Delta t_{i+1}]\}$ are determined according to Assumption 5. Followability is achieved via (8g) and via:

Assumption 7. The optimisation problem (8) is feasible.

Lemma 2. Given Assumptions 2, 5, 6, 7, and $\bar{k} \geq T_\text{p}/T_\text{s}$ the posterior mean (3) of a GP trained with (8) is trackable in the sense of Definition 1 for system (1).

Proof 2. Under Assumption 7 optimisation problem (8) guarantees $m^+(t) \in [m^+_{\min}, m^+_{\max}] \subseteq \mathcal{X}$ for all $t \in [0, T_\text{s}\bar{k}]$. With $\bar{k} \geq T_\text{p}/T_\text{s}$ and Assumptions 2 and 6 $m^+(t + nT_\text{p}) = m^+(t)$, where $n \in \mathbb{N}$ and consequently $m^+(t) \in \mathcal{X}$ for all $t \in \mathbb{R}$. For each time interval $\Delta t_i \geq t \geq \Delta t_{i+1}$, $\dot{m}^+(t) \in [m^+_{\min,i}, m^+_{\max,i}] \subseteq [\underline{\tau}_i, \overline{\tau}_i]$, such that with Assumption 5 $m^+(t) \in \mathcal{T}$ for each $t \in [0, T_\text{p}/T_\text{s}]$. With $\bar{k} \geq T_\text{p}/T_\text{s}$ and Assumptions 2 and 6 $m^+(t) \in \mathcal{T}$ for all $t \in \mathbb{R}$, and consequently $m^+(t) \in (\mathcal{X} \cap \mathcal{T})$ for all $t \in \mathbb{R}$. □

Please note that from a practical side, optimisation problem (4) with very small sampling time T_s might be preferred over (8) as it is computationally cheaper and tends to the same result for $T_\text{s} \to 0$.

Example 2. Given is the dynamical system $x(k+1) = 0.9x(k) + 0.1u(k)$ with state constraints $\mathcal{X} = [-2, 2]$ and input constraints $\mathcal{U} = [-3, 1.4]$. In accordance to Assumptions 2 and 6 a zero prior mean function $m(t_*) = 0$ and a periodical covariance function $\kappa = \theta_1^2 \exp(-\frac{2}{\theta_2^2} \sin(\pi\theta_3^{-1}(t_* - t_i)^2))$ are chosen. Figure 6 shows the (unknown) reference $r(k)$ in solid black which should be modelled based on the observations depicted as black crosses. The predicted mean should be consistent with $\mathcal{X} \cap \mathcal{T}_k$, depicted as dark grey area. The constrained Gaussian process provides a reference prediction depicted as dashed

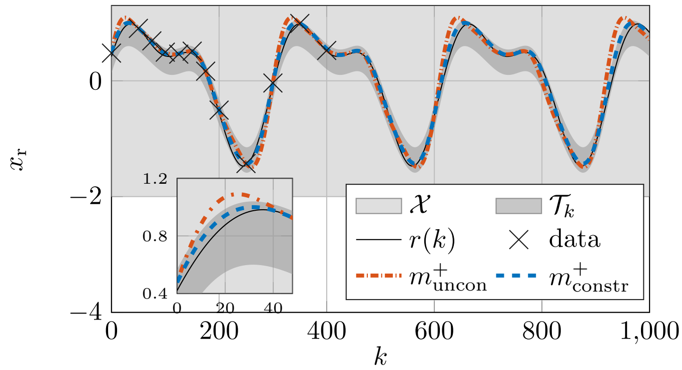

Fig. 6. GP prediction of a periodic reference. The underlying unknown reference $r(k) = \sin(2k) + 0.5\sin(4k+1)$ is modelled based on the data points ($\times$).

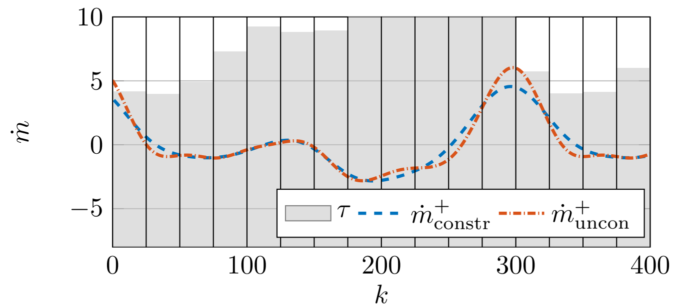

Fig. 7. Inner tube growth rate approximation (shaded area with $\eta = 16$ intervals) should not be left by $\dot{m}$.

line. It satisfies the trackability conditions for all times. In contrast to this, an unconstrained hyperparameter optimisation using the same training data leads to the red dash-dotted prediction which violates the reachable tube. This can be seen e.g. in the inlay plot in Figure 6 showing a zoomed view on the first 50 steps as well as in Figure 7. The derivative of the unconstrained mean $\dot{m}^+_{\text{uncon}}$ violates the inner approximation of the tube growth rate (the trajectory leaves the grey area) depicted in Figure 7. In contrast, the derivative of the constrained mean $\dot{m}^+_{\text{constr}}$ is fulfilling those constraints. Even though the unconstrained GP satisfies the state constraints $\mathcal{X}$, it is not trackable by the system dynamics due to the input constraints.

4. CONCLUSION

We outlined how Gaussian processes can model external reference signals of different structure (asymptotically constant or periodic). These GP predictions can be used e.g. in model predictive control to achieve an improved tracking performance as well as providing stability guarantees. To do so, we have proposed different algorithms to train GPs. These concepts are based on constrained hyperparameter optimisation to guarantee trackability and constrained satisfaction of the predicted GP mean. Investigations for arbitrary references, online learning, and truncated multinormal distributions are interesting future research directions.

REFERENCES

Berkenkamp, F. and Schoellig, A.P. (2015). Safe and robust learning control with Gaussian processes. In *European Control Conference (ECC)*, 2496–2501. IEEE.

Blanchini, F. and Miani, S. (2008). *Set-theoretic methods in control*. Springer.

Da Veiga, S. and Marrel, A. (2012). Gaussian process modeling with inequality constraints. In *Annales de la Faculté des sciences de Toulouse: Mathématiques*, volume 21, 529–555.

Faulwasser, T. and Findeisen, R. (2011). A model predictive control approach to trajectory tracking problems via time-varying level sets of Lyapunov functions. In *European Control Conference (ECC)*, 3381–3386.

Ferramosca, A., Limón, D., Alvarado, I., Alamo, T., and Camacho, E. (2009). MPC for tracking of constrained nonlinear systems. In *Conference on Decision and Control (CDC)*, 7978–7983.

Garone, E., Di Cairano, S., and Kolmanovsky, I. (2017). Reference and command governors for systems with constraints: A survey on theory and applications. *Automatica*, 75, 306–328.

Klenske, E.D., Zeilinger, M.N., Schölkopf, B., and Hennig, P. (2016). Gaussian process-based predictive control for periodic error correction. *IEEE Transactions on Control Systems Technology*, 24(1), 110–121.

Kocijan, J. (2016). *Modelling and control of dynamic systems using Gaussian process models*. Springer.

Kocijan, J., Murray-Smith, R., Rasmussen, C., and Girard, A. (2004). Gaussian process model based predictive control. In *American Control Conf.*, 2214–2219.

Limon, D., Alamo, T., de la Peña, D.M., Zeilinger, M.N., Jones, C., and Pereira, M. (2012). MPC for tracking periodic reference signals. In *Nonlinear Model Predictive Control Conference (NMPC)*, 490–495.

Limon, D., Alvarado, I., Alamo, T., and Camacho, E. (2008). MPC for tracking piecewise constant references for constrained linear systems. *Automatica*, 44(9), 2382–2387.

Maiworm, M., Wagner, C., Temirov, R., Tautz, F.S., and Findeisen, R. (2018). Two-degree-of-freedom control combining machine learning and extremum seeking for fast scanning quantum dot microscopy. In *American Control Conference (ACC)*, 4360–4366.

Matschek, J., Bäthge, T., Faulwasser, T., and Findeisen, R. (2019). Nonlinear predictive control for trajectory tracking and path following: An introduction and perspective. In *Handbook of Model Predictive Control*, 169–198. Springer.

Matschek, J., Gonschorek, T., Hanses, M., Elkmann, N., Ortmeier, F., and Findeisen, R. (2020). Learning references with Gaussian processes in model predictive control applied to robot assisted surgery. In *European Control Conference (ECC)*. To appear.

Ostafew, C.J., Schoellig, A.P., Barfoot, T.D., and Collier, J. (2016). Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking. *Journal of Field Robotics*, 33(1), 133–152.

Rasmussen, C.E. and Williams, C.K.I. (2006). *Gaussian Processes for machine learning*. MIT Press.

Rawlings, J.B., Mayne, D.Q., and Diehl, M. (2017). *Model predictive control: theory, computation, and design*, volume 2. Nob Hill Publishing Madison, WI.

Salzmann, M. and Urtasun, R. (2010). Implicitly constrained Gaussian process regression for monocular non-rigid pose estimation. In *Advances in Neural Information Processing Systems*, 2065–2073.