Proceedings of Machine Learning Research vol 120:1–10, 2020 2nd Annual Conference on Learning for Dynamics and Control

Learning supported Model Predictive Control for Tracking of Periodic References

Janine Matschek JANINE.MATSCHEK@OVGU.DE
Laboratory for Systems Theory and Automatic Control, Otto von Guericke University Magdeburg

Rolf Findeisen ROLF.FINDEISEN@OVGU.DE
Laboratory for Systems Theory and Automatic Control, Otto von Guericke University Magdeburg

Editors: A. Bayen, A. Jadbabaie, G. J. Pappas, P. Parrilo, B. Recht, C. Tomlin, M. Zeilinger

Abstract

Increased autonomy of controllers in tasks with uncertainties stemming from the interaction with the environment can be achieved by incorporation of learning. Examples are control tasks where the system should follow a reference which depends on measurement data from surrounding systems as e.g. humans or other control systems. We propose a learning strategy for Gaussian processes to model, filter and predict references for model predictive control. Constraints in the learning are included to achieve safety guarantees, enable trackability, and recursive feasibility. An illustrative simulation example for motion compensation shows performance improvements besides the provided guarantees.

Keywords: machine learning, model predictive control, tracking, Gaussian process

1. Introduction

Often references signals that should be tracked by control are provided by the interaction with other systems. Examples are coupled chemical reactors in which the reference of one process depends on the operation of the other (downstream) processes, the cooperative manipulation of objects by multiple robots or robots and humans, as well as an autonomous car which should follow a leading vehicle while keeping constant distance. In all these cases, the reference might be obtained via communication and measurements of the surrounding entities. However, data loss or measurement noise might occur leading to corrupted references. To hedge against these issues, models of the expected reference evolution can be incorporated. Besides operating as fall back, models of the reference allow to predict likely reference evolutions into the future. Especially in advanced control strategies such as model predictive control, these future reference predictions are highly valuable for improved performance, recursive feasibility, constrained satisfaction, and stability. One crucial issue is the question if the reference can actually be followed by the controlled system. Arbitrarily changing references might not be trackable either due to system dynamics, for instance in non-holonomic systems, or due to physical constraints of the actuators or system states. Additionally, noise corrupted references should not be given directly to a controller. They should rather be filtered or smoothed to prevent erroneous behaviour and allow for trackability. In this work we want to address these issues, i.e. prediction, trackability, and filtering of references for control via machine learning. More specific, we want to use a Gaussian process (GP) to model and predict references based on measurements while ensuring that they are trackable. The learned reference is then provided to a tracking model predictive control (MPC) scheme, cf. Figure 1. If the model predictive controller is initially feasible, recursive feasibility with respect to the adjusted reference

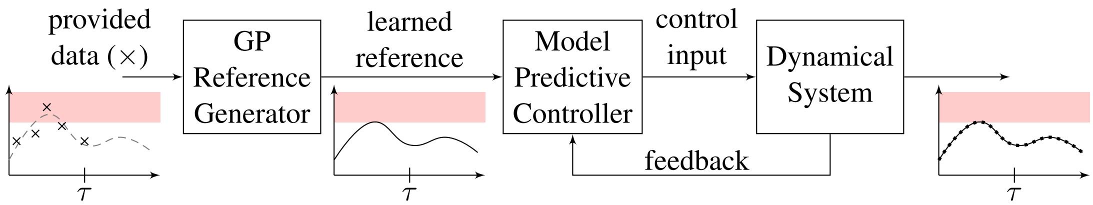


Figure 1: Based on data, denoted by $\times$, the unknown reference (dashed) is modelled by a GP. The GP learns a trackable reference (solid) while prediction ($t > \tau$) and satisfaction of constraints (red) is achieved. The MPC steers the system to exactly follow the reference (dotted).

can be guaranteed due to the design of the learned reference. With respect to enforcing constraint satisfaction and reachability this work is related to reference governors. Reference governors can be viewed as pre-filters that modify a given reference to avoid a constraint violation of the controlled system, see e.g (Garone et al., 2017) for an overview. Our approach also relates to prediction filters, cf. (Jöhl et al., 2020) for a comprehensive comparison of prediction filters for motion compensation. Modification of references in model predictive tracking control is considered in (Limon et al., 2008, 2012; Ferramosca et al., 2009) to achieve recursive feasibility. In these works, artificial references are introduced as additional optimization variables which differ from the original reference if it is not trackable. In (Matschek et al., 2020b) constrained learning of GPs for reference prediction and modification is introduced. Therein, a constrained hyperparameter optimization is utilized to obtain Gaussian process reference prediction models, which guarantee recursive feasibility of a predictive controller. In the present paper, we introduce a modified constrained hyperparameter optimization for periodic references to achieve trackability despite constraints. The benefit of this modified optimization is the elimination of nested optimization problems as proposed in Matschek et al. (2020b). Additionally, a performance comparison of a model predictive controller based on unfiltered, learned and constraint references is provided.

The main contribution of this work is the connection of model predictive control and constrained learning of Gaussian process reference models while providing guarantees. We illustrate our approach by a motion compensation task in robot assisted surgery. In this example, noisy measurements of goal structure motions due to patient breathing are available. A robotic device should follow this motion to eliminate relative motions between surgical instruments and the goal structure. This is achieved by fusing constrained machine learning and model predictive tracking control.

The remainder of this paper is structured as follows: Section 2 outlines the basic principle of GPs and tracking MPC. Section 3 introduces a constrained hyperparameter optimization that generates references suitable for tracking MPC. In Section 4 an example for motion compensation is given. Section 5 concludes this work with a summary and an outlook.

2. Problem formulation

We consider nonlinear, discrete,time systems

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

with continuous system dynamics f and initial conditions $x_0 \in \mathbb{R}^{n_x}$. The state $x \in \mathbb{R}^{n_x}$ and the input $u \in \mathbb{R}^{n_u}$ should satisfy the closed state constraints $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ and the compact input constraint $\mathcal{U} \subseteq \mathbb{R}^{n_u}$, respectively. Each state should follow a time dependent reference, which is a priori unknown and only given in terms of possibly noisy observations $D := \{(t_i, y_i) \in \mathbb{R}_0^+ \times \mathbb{R}^{n_x} \mid i = 1, 2, \ldots, n_D\}$, where t_i are time stamps and y_i are measured reference values at those times. The overall task is the design of an MPC to track a reference that is modelled based on these observations. The MPC steers the system to follow the reference while considering future predictions of both the system states and the time dependent reference. The GP reference model predicts upcoming reference values while taking additional limitations and constraints into account. These constraints are formed to guarantee trackability of the reference:

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

Arbitrarily changing references can lead to infeasibility of the optimal control problem if the terminal constraints and the reachable set do not overlap, see e.g. (Limon and Alamo, 2015). Enforcing trackability of the learned references is sufficient to maintain feasibility of optimal control problems with terminal equality constraints in the nominal case when they are initially feasible. In this case, trackability ensures the existence of a feasible input independently from the length of the prediction horizon of the MPC. Extensions from terminal equality to inequality constraints were derived in (Faulwasser and Findeisen, 2011; Faulwasser, 2013) to guarantee convergence of tracking MPC, which implicitly also assumes trackability to obtain suitable reference inputs. To guarantee trackability, we propose to use machine learning, i.e. Gaussian processes with a specific learning algorithm. A brief introduction to GPs as well as to MPC for tracking is given in the following.

2.1. Gaussian Processes as Reference Generators

Gaussian processes are stochastic system identification tools, which recently gained popularity also in control problems, see e.g. (Ostafew et al., 2016; Kocijan et al., 2004; Berkenkamp and Schoellig, 2015; Klenske et al., 2016; Maiworm et al., 2018; Matschek et al., 2020a; Pöhler et al., 2019). They are extensions of Gaussian distributions to the function space such that each finite dimensional subset of this infinite-dimensional distribution builds a joint Gaussian distribution (Rasmussen and Williams, 2006). GPs can be described by a mean function $m : \mathbb{R} \to \mathbb{R}$ and a symmetric, positive semi-definite covariance function $\kappa : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_0^+$. We denote a Gaussian process 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 output of a GP $y(t)$ at each specific input t is normal distributed. In this setup, we will use GPs to map from the time domain to the reference and use multiple uncorrelated GPs for systems with $n_x > 1$. Moreover, we will use the posterior mean as the learned reference, i.e. $x_\mathrm{r}(k) = m^+(T_\mathrm{s} k)$, where T_s is the sampling time. The goal is to learn posterior mean values m^+ which predict suitable references fulfilling state constraints and ensure trackability. Following along the lines of Matschek et al. (2020b) we want to predict the reference for current and future times t_* based on observations which build a training data set $D_\mathrm{t} := \{\boldsymbol{t}, \boldsymbol{y}\} \subseteq D$, with

$$\boldsymbol{t} := [t_{\mathrm{t},1}, \ldots, t_{\mathrm{t},n_{D_\mathrm{t}}}], \quad \boldsymbol{y} := [y_{\mathrm{t},1}, \ldots, y_{\mathrm{t},n_{D_\mathrm{t}}}], \quad \boldsymbol{m}(\boldsymbol{t}) := [m(t_{\mathrm{t},1}), \ldots, m(t_{\mathrm{t},n_{D_\mathrm{t}}})].$$

Based on a joint probability distribution between training data and unseen test points t_*, the conditional posterior distribution is calculated and the posterior mean $m^+ : \mathbb{R} \to \mathbb{R}$ is given by

$$m^+(t_*) := m(t_*) + K(t_*, \boldsymbol{t})(K(\boldsymbol{t}, \boldsymbol{t}) + \sigma_n^2 I)^{-1}\big(\boldsymbol{y} - \boldsymbol{m}(\boldsymbol{t})\big). \tag{2}$$

Here, K is the covariance matrix whose entries are calculated based on the prior 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 the training data points, while $K(t_*, \boldsymbol{t})$ (with dimension $1 \times n_{D_\mathrm{t}}$) defines the cross correlation between test and training data. The variable σ_n^2 represents the variance of the measurement noise. Periodical references can be represented by the posterior mean of a GP given the following assumption:

Assumption 1 *The prior mean function m is constant. The prior covariance function κ is stationary and periodic, such that $\kappa(t, t') = \kappa(t, t' + nT_\mathrm{p})$ with $n \in \mathbb{N}_0$ and period T_p.*

These prior mean and covariance function depend on so called hyperparameters θ. Besides, they depend on the training data D_t. Whenever beneficial, we will denote this dependency by $m^+(t_*|D_\mathrm{t}, \theta)$. One way to obtain the hyperparameters is the maximization of the logarithmic marginal likelihood function, which results in the most probable hyperparameters given the hyperparameter optimization data $D_\theta := \big\{ (t_{\theta,i}, y_{\theta,i}) \in \mathbb{R}_0^+ \times \mathbb{R}^{n_x} \,|\, i = 1, 2, \ldots, n_{D_\theta} \big\} = \{\boldsymbol{t}_\theta, \boldsymbol{y}_\theta\} \subseteq D$. In (Matschek et al., 2020b) a constrained optimization of the likelihood is proposed, where the predicted mean of the GP is constrained to lie inside the state constraints and a reachable set. This way, trackability of the predicted mean is achieved. In Section 3, an alternative formulation of the constrained hyperparameter optimization for periodic references is proposed, eliminating the drawback of nested optimization problems. The posterior mean m^+ evaluated at discrete time instants $T_\mathrm{s}k$ with $k \in \mathbb{N}$ will be used as the predicted reference for the tracking controller, i.e. $x_\mathrm{r}(k) = m^+(T_\mathrm{s}k)$.

2.2. Tracking Model Predictive Control

Model predictive control is an optimization based control strategy which predicts states via a model of the system. It directly handles constraints during optimization and can be used for nonlinear, multi-input-multi output systems (Mayne et al., 2009; Rawlings et al., 2017). In tracking MPC, the control task is to follow a time-dependent reference, cf. (Limon and Alamo, 2015; Matschek et al., 2019; Berberich et al., 2019). If this reference is known to the controller it can predict and minimise the tracking error over its prediction horizon. This will in general lead to an improved tracking performance compared to the case when the future reference evolution is unknown or not used. In our setup, we use GPs to model, filter and predict upcoming references to make them available to tracking MPC. Tracking MPC can be formulated via the optimal control problem

$$\underset{\bar{u}_k(\cdot)}{\text{minimise}} \sum_{i=k}^{k+N-1} L\left(\bar{e}(i), \bar{w}(i)\right) + E\left(\bar{e}(k+N)\right). \tag{3a}$$

subject to $\forall i \in \{k, k+1, \ldots, k+N-1\}$

$$\bar{x}(i+1) = f(\bar{x}(i), \bar{u}_k(i)), \quad \bar{x}(k) = x(k), \tag{3b}$$

$$\bar{e}(i) = x_\mathrm{r}(i) - \bar{x}(i), \tag{3c}$$

$$\bar{w}(i) = u_\mathrm{r}(i) - \bar{u}_k(i), \tag{3d}$$

$$\bar{x}(i) \in \mathcal{X}, \; \bar{u}_k(i) \in \mathcal{U}, \tag{3e}$$

$$\bar{x}(k+N) \in \mathcal{F}(k+N) \subseteq \mathcal{X}. \tag{3f}$$

Here, the system dynamic model for prediction is incorporated via the constraint (3b), where predictions are denoted by a bar over the respective variables. The control errors e and w for the state and input are defined in (3c) and (3d), respectively, where the variable u_r describes the reference input which belongs to state reference x_r. The errors e and w are minimised via the cost functional (3a) consisting of the stage cost L and the terminal cost E. State and input constraints are incorporated into the minimisation via (3e). It is important to note, that the terminal constraint set $\mathcal{F}$ given in (3f) is time dependent. The terminal ingredients E and $\mathcal{F}$ which are normally used to prove stability for MPC depend on the reference to be tracked. As this reference is time-dependent, this time-dependency is also visible in the MPC formulation. For a detailed discussion on differences between setpoint stabilisation and tracking see e.g. (Matschek et al., 2019; Faulwasser and Findeisen, 2011; Faulwasser, 2013). One trivial choice to obtain a stable closed loop is to use terminal equality constraints $x(k+N) = x_\text{r}(k+N)$ and no terminal cost. Obviously, the reference x_r must be known at least for N steps into the future to do so. Additionally, it must satisfy the state constraints $x_\text{r} \in \mathcal{X}$ and must be reachable in N steps as otherwise the optimal control problem is infeasible. Though restrictive, trackability as defined in Definition 1 is sufficient to satisfy these requirements, i.e. it allows for recursive feasibility given initial feasibility independently from the prediction horizon. By using constrained Gaussian process learning we guarantee that the reference is known (via predictions) and trackable. This allows free choice of the MPC prediction horizon, i.e. short horizons do not lead to infeasibility of the optimal control problem. Though constraint satisfaction in the nominal case is guaranteed already by the learned reference, MPC allows satisfaction of the constraints even if additional disturbances, numerical errors or uncertainties appear. This is of mayor importance since operation close to the boundary of the constraints can be demanded.

3. Constrained Hyperparameter Estimation for Periodic References

The reference x_r for the predictive controller is learned based on data D via a Gaussian process. The idea is to use a constrained hyperparameter estimation in which the predicted mean, its derivative and the hyperparameters are constrained to guarantee trackability of the reference. As we consider periodical references but evaluate those constraints only at discrete time points, a satisfaction of constraints for all times from a limited number of evaluation points can be concluded if the period length of the learned reference is an integer multiple of the sampling time. To this end, we constrain the hyperparameter which reflects the period to be a multiple of the sampling time T_s. This results in a mixed integer nonlinear program (MINLP). Moreover we constrain the predicted mean to lie inside the state constraints and its derivative to lie inside constraints reflecting trackability.

We use the one-step reachable tube which is constructed via one-step reachable sets starting at the current reference value, see e.g. (Blanchini and Miani, 2008). The tube is defined as $\mathcal{T}_{k+1} := \mathcal{R}_1(x_\text{r}(k))$, where $\mathcal{R}_1$ defines the set which is reachable in one step starting at $x_\text{r}(k) = m^+(T_\text{s}k)$ which is the predicted reference value. An illustration of the reachable sets is given in Figure 2. The reference is trackable if $x_\text{r}(k) \in (\mathcal{X} \cap \mathcal{T}_k)$ for all $k \in \mathbb{N}$. Instead of using these sets directly, we use the maximum and minimum possible rate of change $\overline{\tau}$ and $\underline{\tau}$, which are also depicted in Figure 2. Based on the one-step reachable set, the rates $\overline{\tau}$ and $\underline{\tau}$ can be expressed as functions of the mean m^+. If the change $\dot{m}^+(T_\text{s}k)$ of the predicted reference lies inside the set $[\underline{\tau}(T_\text{s}k), \overline{\tau}(T_\text{s}k)]$ and $m^+(T_\text{s}k) \in \mathcal{X}$ for all k then the system can follow the reference once starting on it.

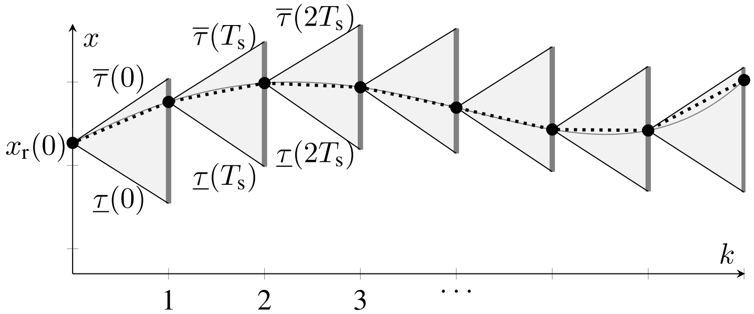


Figure 2: The posterior mean m^+ (solid grey line) is evaluated at discrete times to build the reference x_r (black dots). The reachable sets starting at $x_r(k)$ are illustrated by thick grey vertical lines at each consecutive step $k+1$. The slopes of the triangles spanning the reachable set are $\underline{\tau}$ and $\overline{\tau}$. They build the constraints for $\dot{m}^+$ (black dotted line).

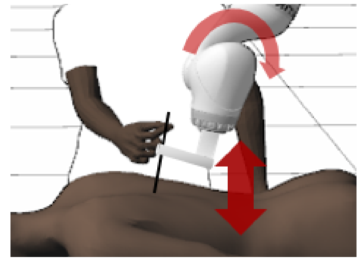


Figure 3: Illustration of the application example: A robot assist a physician during spine surgery while compensating breathing motions via GP supported tracking MPC.

The resulting optimization problem can be written as

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

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

$$\dot{m}^+(t_*|D_\theta,\theta) \in [\underline{\tau}(t_*), \overline{\tau}(t_*)], \tag{4c}$$

$$t_* = T_s k,\ \forall k \in \left\{0,\dots,\bar{k}\right\}, \tag{4d}$$

$$\tilde{T}_p(\theta) = T_s q,\ \text{with } q \in \mathbb{N}, \tag{4e}$$

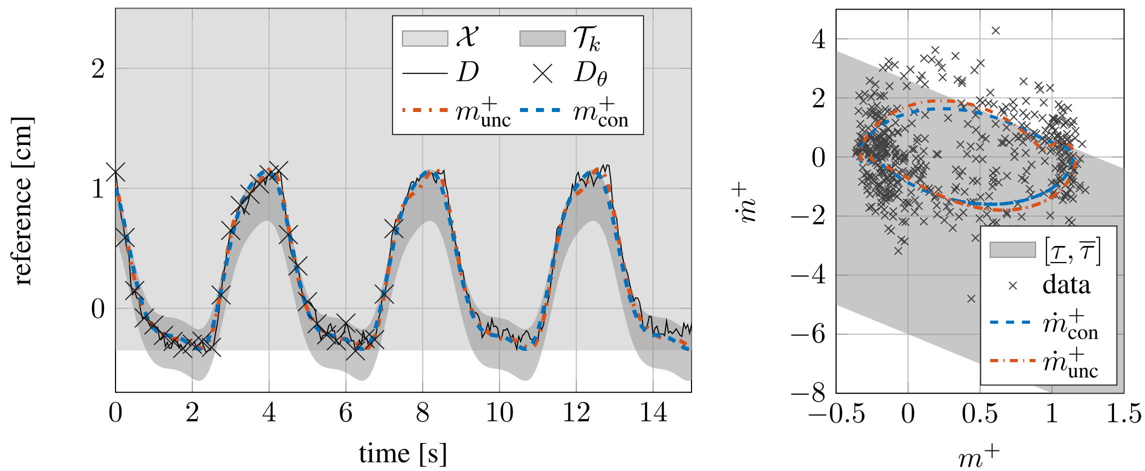
where the cost function $l(\theta)$ is the negative logarithmic marginal likelihood

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

In contrast to (Matschek et al., 2020b) the estimated period length $\tilde{T}_p$, which is a hyperparameter, is constrained to be an integer multiple of T_s. This allows evaluation of the constraints at discrete times only. The derivative of the mean is calculated via $\dot{m}^+(T_s k) := (m^+(T_s(k+1)) - m^+(T_s k))\,T_s^{-1}$. In general, this mixed integer problem is computational challenging despite the low number of optimization variables and the avoidance of nested optimizations since the training data matrix inversion scales with $\mathcal{O}(n_{D_\theta}^3)$. However, fast execution times are of secondary importance, as we consider offline learning. Under the assumption of feasibility of (4) we can state the following:

Lemma 1 *Given Assumption 1, and $\bar{k} \geq \tilde{T}_p/T_s$ the posterior mean (2) of a GP trained with (4) is trackable in the sense of Definition 1 for system (1).*

Proof Assuming feasibility, problem (4) guarantees $m^+(kT_s) \in \mathcal{X}$ for all $k \in \{0,\dots,\bar{k}\}$. Furthermore, $\dot{m}^+(kT_s) \in [\underline{\tau}(kT_s), \overline{\tau}(kT_s)]$ for all $k \in \{0,\dots,\bar{k}\}$. This implies that $m^+(kT_s) \in \mathcal{T}_k$ and consequently $m^+(kT_s) \in \mathcal{X} \cap \mathcal{T}_k$ for all $k \in \{0,\dots,\bar{k}\}$. Assumption 1 makes it possible to infer $m^+(kT_s + i\tilde{T}_p) = m^+(kT_s) \in \mathcal{X}$ with $i \in \mathbb{N}$ for all $k \in \{0,\dots,\bar{k}\}$. Since $\bar{k} \geq \tilde{T}_p/T_s$ and $\tilde{T}_p = qT_s$ the state constraints $m^+(kT_s) \in \mathcal{X}$ are fulfilled for all $k \in \mathbb{N}$. Similar reasoning applies to $m^+(kT_s) \in \mathcal{T}_k$, such that $m^+(kT_s) \in (\mathcal{X} \cap \mathcal{T}_k)$ for all $k \in \mathbb{N}$. ■



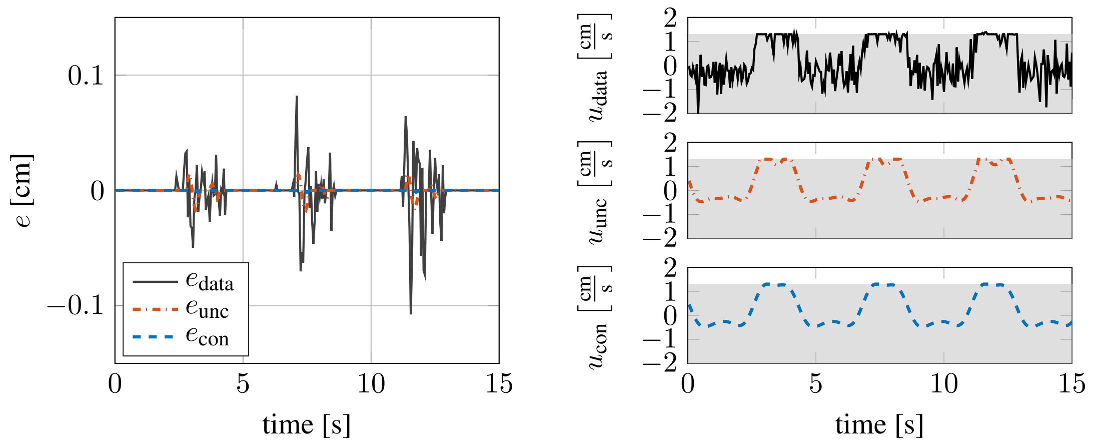
(*a*) Prediction of reference via a constrained GP (blue, dashed) and an unconstrained (red, dash-dotted) one.

(*b*) Derivative of the predicted mean.

Figure 4: Constrained Gaussian process reference learning.

4. Application Example

To illustrate the proposed approach we apply the constrained learning and tracking of references via MPC in a medical robotics task. In robot supported interventions, robots should for example hold devices in specific positions relative to a goal structure. These goal structures are not necessary fixed in space but change due to patient motions. Periodic motions can e.g. occur due to breathing. These motions are often measured via external tracking systems as e.g. stereo cameras. We will use the proposed learning algorithm for Gaussian processes to learn these motions and follow them with a surgical instrument held by the robot via tracking MPC. In this example we consider vertical motions of a goal structure due to breathing which should be compensated by a robot equipped with a linear motor as a tool. For simplicity of presentation, we approximate the dynamic model of the tool by a linear one-dimensional integrator for motions in vertical direction. Clinical data D was used for the motion compensation and is depicted in Figure 4(*a*) in black solid line. The sampling time is $T_\text{s} = 0.05s$. From these measurements a training data set $D_\theta \subset D$ was formed, depicted as black crosses. The prior mean $m = 0$ and prior covariance function $\kappa = \theta_1^2 \exp(-\frac{2}{\theta_2^2}\sin(\pi\theta_3^{-1}(t_* - t_i)^2))$ are chosen. The hyperparameter optimization was performed while taking into account the state constraints $\mathcal{X} = [-0.35, 20]$ depicted as light grey shaded area. These state constraints are also used in the predictive controller. The lower bound of $\mathcal{X}$ was introduced to prevent the patient from harm by a too deeply inserted surgical device held by the robot and is therefore crucial for safety reasons. The measurements however slightly violate this constraint due to sensor noise at $t = 6.25s$. Furthermore, the constraints induced by the reachable tube (depicted as dark grey shaded area) should be fulfilled. To this end the derivative of the predicted mean is calculated and shown in Figure 4(*b*). The feasible set spanned by the bounds $\underline{\tau}, \overline{\tau}$ for the derivative is depicted as grey area. Given $\underline{\tau}, \overline{\tau}$, and $\mathcal{X}$ a constrained hyperparameter optimization via (4) was performed resulting in the predicted mean m^+_{con} depicted in blue dashed line in Figure 4(*a*). The corresponding derivative $\dot{m}^+_{\text{con}}$ is depicted in Figure 4(*b*) in blue dashed line. All constraints are satisfied leading to a trackable reference. In contrast, an unconstrained hyperparameter optimization was performed



(*a*) Tracking errors e_{data}, e_{unc}, and e_{con} for the three different references.

(*b*) Optimal control inputs u_{data}, u_{unc}, and u_{con} to follow the three different references.

Figure 5: Tracking MPC performance for three different references based on the unfiltered measurement data, an unconstrained GP and a GP with constrained learning.

leading to m^+_{unc} and $\dot{m}^+_{\text{unc}}$ depicted in red dash-dotted line in Figures 4(*a*) and 4(*b*). The derivative $\dot{m}^+_{\text{unc}}$ violates the upper bound of the reachability constraints, cf. Figure 4(*b*). The unfiltered data is also not trackable as it violates both the state constraints and the derivative constraints, cf. 4(*a*) and 4(*b*). To show the effect of the trackability, all three references, i.e. the original noisy data, the unconstrained GP and the constrained GP, are used in the same model predictive controller with a prediction horizon of $N = 10$. The tracking errors and the optimal input are depicted in Figure 5. The largest tracking error occurs when unfiltered data is directly used as reference, see Figure 5(*a*) (black solid line). As this reference is not trackable even infeasibility of the optimal control problem occurred at several times. Furthermore, the control input is very noisy, cf. Figure 5(*b*), which is undesirable in many respects (e.g. due to wear and tear or physical limits). A GP learned on the training data without constraints in the hyperparameter optimization produces a smoother reference. The tracking errors are significantly smaller but trackability of the reference is not given, such that input constraints become active and the control error rises at those times, cf. Figure 5 (red dash-dotted lines). In contrast, the reference generated with constrained GP learning is trackable and shows the smallest control errors which solely origin from numerics, cf. Figure 5(*a*), dashed line.

5. Conclusion

As shown, constrained Gaussian process learning based on available data can be used to model, predict, and filter references, while guaranteeing trackability of the reference. The proposed algorithm for constrained Gaussian process learning for periodic references reduces computational complexity compared to similar approaches by avoiding nested optimisations. The enforced trackability ensures recursive feasibility of a model predictive controller that steers the system to follow the reference. In addition to the derived guarantees, an improvement in performance can be achieved, as underlined in an application example considering robotic motion compensation. Extensions to arbitrary references as well as online learning are interesting future fields of research.

References

Julian Berberich, Johannes Köhler, Matthias Müller, and Frank Allgöwer. Data-driven tracking MPC for changing setpoints. *arXiv preprint arXiv:1910.09443*, 2019.

Felix Berkenkamp and Angela Schoellig. Safe and robust learning control with Gaussian processes. In *European Control Conference (ECC)*, pages 2496–2501, 2015.

Franco Blanchini and Stefano Miani. *Set-theoretic methods in control*. Springer, 2008.

Timm Faulwasser. *Optimization-based Solutions to Constrained Trajectory-tracking and Path-following Problems*. Number 3 in Contributions in Systems Theory and Automatic Control. Shaker Verlag, Otto-von-Guericke University Magdeburg, 2013.

Timm Faulwasser and Rolf Findeisen. A model predictive control approach to trajectory tracking problems via time-varying level sets of lyapunov functions. In *Joint Conference on Decision and Control (CDC) and European Control Conference (ECC)*, pages 3381–3386, 2011.

Antonio Ferramosca, Daniel Limon, Ignacio Alvarado, Teodoro Alamo, and Eduardo Camacho. MPC for tracking of constrained nonlinear systems. In *Joint Conference on Decision and Control (CDC) and Chinese Control Conference*, pages 7978–7983, 2009.

Emanuele Garone, Stefano Di Cairano, and Ilya Kolmanovsky. Reference and command governors for systems with constraints: A survey on theory and applications. *Automatica*, 75:306–328, 2017.

Alexander Jöhl, Stefanie Ehrbar, Matthias Guckenberger, Stephan Klöck, Mirko Meboldt, Melanie Zeilinger, Stephanie Tanadini-Lang, and Marianne Schmid Daners. Performance comparison of prediction filters for respiratory motion tracking in radiotherapy. *Medical physics*, 47(2):643–650, 2020.

Edgar Klenske, Melanie Zeilinger, Bernhard Schölkopf, and Philipp Hennig. Gaussian process-based predictive control for periodic error correction. *IEEE Transactions on Control Systems Technology*, 24(1):110–121, 2016.

Juš Kocijan, Roderick Murray-Smith, Carl Edward Rasmussen, and Agathe Girard. Gaussian process model based predictive control. In *American Control Conference (ACC)*, volume 3, pages 2214–2219, 2004.

Daniel Limon and Teodoro Alamo. Tracking Model Predictive Control. In John Baillieul and Tariq Samad, editors, *Encyclopedia of Systems and Control*, pages 1475–1484. Springer, 2015.

Daniel Limon, Ignacio Alvarado, Teodoro Alamo, and Eduardo F Camacho. MPC for tracking piecewise constant references for constrained linear systems. *Automatica*, 44(9):2382–2387, 2008.

Daniel Limon, Teodoro Alamo, David Muñoz de la Peña, Melanie Zeilinger, Colin Jones, and Mario Pereira. MPC for tracking periodic reference signals. In *Nonlinear Model Predictive Control Conference (NMPC)*, pages 490–495. Elsevier, 2012.

Michael Maiworm, Christian Wagner, Ruslan Temirov, F Stefan Tautz, and Rolf Findeisen. Two-degree-of-freedom control combining machine learning and extremum seeking for fast scanning quantum dot microscopy. In *American Control Conference (ACC)*, pages 4360–4366, 2018.

Janine Matschek, Tobias Bäthge, Timm Faulwasser, and Rolf Findeisen. Nonlinear predictive control for trajectory tracking and path following: An introduction and perspective. In *Handbook of Model Predictive Control*, pages 169–198. Springer, 2019.

Janine Matschek, Tim Gonschorek, Magnus Hanses, Norbert Elkmann, Frank Ortmeier, and Rolf Findeisen. Learning references with Gaussian processes in model predictive control applied to robot assisted surgery. In *European Control Conference (ECC)*, 2020a. to appear.

Janine Matschek, Andreas Himmel, Kai Sundmacher, and Rolf Findeisen. Constrained Gaussian process learning for model predictive control. In *IFAC World Congress*, 2020b. to appear.

David Mayne, Saša Raković, Rolf Findeisen, and Frank Allgöwer. Robust output feedback model predictive control of constrained linear systems: Time varying case. *Automatica*, 45(9):2082–2087, 2009.

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

Lukas Pöhler, Jonas Umlauft, and Sandra Hirche. Uncertainty-based human motion tracking with stable Gaussian process state space models. In *Conference on Cyber-Physical and Human Systems (CPHS)*, pages 8–14. Elsevier, 2019.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for machine learning*. MIT Press, 2006.

James Rawlings, David Mayne, and Moritz Diehl. *Model predictive control: theory, computation, and design*, volume 2. Nob Hill Publishing Madison, WI, 2017.