

# KPC: Learning-Based Model Predictive Control with Deterministic Guarantees

Emilio T. Maddalena<sup>1</sup>

Paul Scharnhorst<sup>1,2</sup>

Yuning Jiang<sup>1</sup>

Colin N. Jones<sup>1</sup>

EMILIO.MADDALENA@EPFL.CH

PAUL.SCHARNHORST@EPFL.CH

YUNING.JIANG@EPFL.CH

COLIN.JONES@EPFL.CH

<sup>1</sup>*École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland*

<sup>2</sup>*Swiss Center for Electronics and Microtechnology, Neuchâtel, Switzerland*

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

## Abstract

We propose Kernel Predictive Control (KPC), a learning-based predictive control strategy that enjoys deterministic guarantees of safety. Noise-corrupted samples of the unknown system dynamics are used to learn several models through the formalism of non-parametric kernel regression. By treating each prediction step individually, we dispense with the need of propagating sets through highly non-linear maps, a procedure that often involves multiple conservative approximation steps. Finite-sample error bounds are then used to enforce state-feasibility by employing an efficient robust formulation. We then present a relaxation strategy that exploits on-line data to weaken the optimization problem constraints while preserving safety. Two numerical examples are provided to illustrate the applicability of the proposed control method.

**Keywords:** Model Predictive Control, Safe Learning, Deterministic Guarantees, Non-Parametric Kernel Regression.

## 1. Introduction

Safety is the number one requirement for any system that operates under physical constraints. For decades, this has been a major concern when control systems incorporate forms of adaptation or learning (Anderson, 2005; Garcia and Fernández, 2015; Hewing et al., 2020; Wabersich and Zeilinger, 2020). A considerable body of literature exists establishing stability and performance guarantees in scenarios of *parametric* plant-model mismatch (see Lorenzen et al. (2019); Tanaskovic et al. (2019); Bujarbaruah and Vallon (2020) for some recent works in this direction). Depending on the final application however, assuming that the exact model structure is available might be unrealistic due to the complex physics behind the system at hand, or to the time-monetary costs associated to the modeling process.

A compelling alternative to the paradigm described above is the use of non-parametric models. These form a flexible class of surrogate functions whose number of parameters grows with the cardinality of the dataset. Relevant examples for the control community include the Nonlinear Set Membership (NSM) (Milanese and Novara, 2004; Maddalena and Jones, 2020) and the Kinky Inference (KI) (Calliess et al., 2020; Manzano et al., 2020) techniques. Due to the ease of incorporating prior expert knowledge and the inherent uncertainty quantification associated with them, Gaussian processes (GPs) have recently become a popular modeling tool for dynamical systems

Capone et al. (2020); Matschek and Findeisen (2020); Arcari et al. (2020); Umlauf and Hirche (2020); Shukla et al. (2020); Lian and Jones (2020). Such function approximators are typically paired with appropriate Model Predictive Control (MPC) schemes that not only take into account the latent function estimate, but also the model variance to act with care in highly uncertain regions of the space (see Koller et al. (2018); Hewing et al. (2019) for two examples, and Beckers et al. (2019) for an exception to this trend).

As opposed to Gaussian processes, kernel ridge regression (KRR) and support vector regression (SVR) are deterministic non-parametric tools. These models have the same form of a GP predictive mean: a weighted sum of kernel basis functions. Moreover, with an appropriate choice of regularization constant, a KRR model matches exactly a GP posterior (Kanagawa et al., 2018). Connections between the stochastic and the deterministic frameworks are profound and have been long known (Kimeldorf and Wahba, 1970). Uncertainty can be quantified in the KRR and SVR cases by considering all maps belonging to their underlying reproducing kernel Hilbert space (RKHS) of functions. The RKHSs associated to various kernels, including the widely used squared-exponential, are dense in the space of continuous functions with compact domains (Micchelli et al., 2006).

**Our contribution:** We propose in this paper a predictive control strategy based on non-parametric kernel regression that incorporates deterministic guarantees of safety. Samples from the unknown ground-truth dynamics are used to construct one-step and multi-step ahead models, with an appropriate state-dependent uncertainty quantification obtained from recently derived error-bounds (Maddalena et al., 2020). The available dataset can be contaminated by noise, which is only assumed to be bounded, but otherwise drawn from any distribution. An efficient robust optimization formulation is derived to enforce state-constraint satisfaction. We then present a relaxation strategy that exploits on-line information to alleviate the problem constraints while preserving safety. Two numerical examples are provided and we discuss scalability issues to large datasets.

## 2. Problem Definition

We consider a discrete-time nonlinear system of the form

$$x_{t+1} = f(x_t, u_t) \quad (1)$$

with time  $t \in \mathbb{N}$ , states  $x_t \in \mathbb{R}^{n_x}$ , inputs  $u_t \in \mathbb{R}^{n_u}$ , and *unknown*<sup>1</sup> transition map  $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_x}$ . Hard constraints  $x_t \in \mathbb{X} = \{x \mid g_i(x) \leq 0, i = 1, \dots, n_{\mathbb{X}}\}$  and  $u_t \in \mathbb{U} = \{u \mid s_i(u) \leq 0, i = 1, \dots, n_{\mathbb{U}}\}$  are imposed for all  $t \in \mathbb{N}$ , where  $\mathbb{X}$  and  $\mathbb{U}$  are polyhedra. More general compact sets could also be considered herein, but the geometric assumptions posed on  $\mathbb{X}$  and  $\mathbb{U}$  will later allow for an efficient robust reformulation as detailed in Section 4. Given a *safe subset* of the state space

$$\mathbb{X}_{\text{safe}} \subseteq \mathbb{X} \quad (2)$$

our goal is to drive the dynamical system (1) from a specified initial condition  $x_0$  to the set  $\mathbb{X}_{\text{safe}}$  while satisfying all constraints. Similarly to Koller et al. (2018), we also assume that a local policy  $\pi_{\text{safe}} : \mathbb{X}_{\text{safe}} \rightarrow \mathbb{U}$  is available, making  $\mathbb{X}_{\text{safe}}$  forward invariant, i.e.,  $\forall x_t \in \mathbb{X}_{\text{safe}} : f(x_t, \pi_{\text{safe}}(x_t)) \in \mathbb{X}_{\text{safe}}, \pi_{\text{safe}}(x_t) \in \mathbb{U}$ . A frequent instance of this problem is the regulation to a specific fixed point, in which  $\mathbb{X}_{\text{safe}} = \{x_{\text{eq}}\}$  and  $\pi(x_{\text{eq}}) = u_{\text{eq}}$  is the equilibrium control constant. In order to accomplish our task, we make use of noise-corrupted measurements of our unknown ground-truth and the formalism of non-parametric kernel learning are described next.

---

1. If a partial model for the latent function is available, the learning task is simply performed on the error dynamics.

### 3. Non-Parametric Kernel Learning

In this section only, we consider  $n_x = 1$  simply to avoid using a cumbersome notation; for  $n_x > 1$ , each output component of  $f$  has to be considered separately. Moreover, the shorthand notation  $f(x, u) = f(z)$  is used. Suppose the map  $f$  is unknown, but a collection of  $D$  measurement pairs is available to reconstruct it

$$\mathbb{D} = \{(z_d, y_d)\}_{d=1}^D \quad \text{where} \quad y_d = f(z_d) + \delta_d \quad (3)$$

In this formulation, we assume the states of the system to be measurable (the same is required for instance in [Koller et al. \(2018\)](#); [Capone et al. \(2020\)](#)). We also make the following two assumptions on our dataset and on the observational model.

**Assumption 1** *The data locations  $z_1, \dots, z_D$  are pairwise distinct.*

**Assumption 2** *The noise affecting all data-points  $\delta = (\delta_1 \dots \delta_D)$  is bounded in module by a known quantity  $\bar{\delta} \geq |\delta|$ ,  $\bar{\delta} \in \mathbb{R}_{\geq 0}^D$ .*

**Remark 1** The core difference between our approach and a GP-based approach lies in Assumption 2. While we assume a bounded quantity, no distribution is imposed on the noise, contrary to the GP case where it is customary to assume Gaussian noise. Due to this disparity, a direct comparison between the two techniques is not possible.

The approach of kernel machines is employed to learn the unknown dynamics from the available dataset. Next, we recall the basics of such theory, see [Schölkopf et al. \(2018\)](#) for a more complete coverage of the topic. A kernel is any real-valued symmetric positive-semidefinite function  $k : \mathcal{Z} \times \mathcal{Z} \rightarrow \mathbb{R}$ . Each kernel defines a reproducing kernel Hilbert space  $\mathcal{H} \subset \mathbb{R}^{\mathcal{Z}}$ , where  $\forall z \in \mathcal{Z}$  we have that  $k(z, \cdot) \in \mathcal{H}$ . Computing the inner-product between a map  $h \in \mathcal{H}$  and a partially evaluated kernel  $k(z, \cdot)$  is equivalent to assessing the value of  $h$  at  $z$ , i.e.,  $\langle h, k(z, \cdot) \rangle_{\mathcal{H}} = h(z)$ , which is known as the reproducing property. Members  $f$  of  $\mathcal{H}$  can be seen as linear combinations of partially evaluated kernel functions since  $\mathcal{H}$  is the closure of  $\text{span}(k(z, \cdot))$ ,  $\forall z \in \mathcal{Z}$  with respect to the induced metric. The norm in the  $\mathcal{H}$  space is defined as  $\|h\|_{\mathcal{H}} = \sqrt{\langle h, h \rangle_{\mathcal{H}}}$ .

For convenience, we define  $Z$  as the collection of all dataset inputs  $z_d$ , and  $y$  as the collection of all targets  $y_d$ . Also, let  $K \in \mathbb{R}^{D \times D}$  be the constant matrix of kernel evaluations at  $Z$ , i.e.,  $k(z_i, z_j)$  at its  $i$ -th row and  $j$ -th column, and let  $K_{Zz} : \mathcal{Z} \rightarrow \mathbb{R}^D$  denote the column vector function  $z \mapsto (k(z_1, z), \dots, k(z_D, z))^{\top}$  and  $K_{zZ}$  simply represents its transpose. Finally, the so-called power function is a non-negative map  $P : \mathcal{Z} \rightarrow \mathbb{R}_{\geq 0}$  defined as

$$P(z) = \sqrt{k(z, z) - K_{zZ} K^{-1} K_{Zz}} \quad (4)$$

and evaluates to zero for all  $z_d$  in the dataset ([Wendland, 2004](#), Sec. 11.1). Note the similarity between the power function  $P(z)$  and the posterior variance of a Gaussian process.

The estimate  $\hat{f}$  is built by minimizing a combination of the mean-squared error and a regularization term to penalize complexity, i.e., the kernel ridge regression (KRR) cost

$$\hat{f} = \arg \min_{h \in \mathcal{H}} \left\{ \frac{1}{D} \sum_{(z,y) \in \mathbb{D}} (h(z) - y)^2 + \lambda \|h\|_{\mathcal{H}}^2 \right\}. \quad (5)$$

According to the well-known representer theorem (Schölkopf et al., 2018), out of all possible maps  $h \in \mathcal{H}$ , a minimizer exists and is given by a weighted sum of kernels centered at the input locations  $Z$ . The problem above is therefore equivalent to a finite-dimensional quadratic program whose closed-form solution, our nominal model, is given by

$$\hat{f} = K_{zz} (K + D\lambda I)^{-1} y \quad (6)$$

**Remark 2** Note that the map described by (6) has the same form as a Gaussian process posterior distribution conditioned on the data, that is, its predictive mean. Indeed, if  $\sigma$  is the noise variance in the GP scenario and  $\lambda$  is selected as  $\sigma^2/D$ , the two models are exactly the same. The reader is referred to (Williams and Rasmussen, 2006, § 6.1) for a discussion on the existing connections.

**Assumption 3** *The chosen kernel  $k(\cdot, \cdot)$  is a strictly positive-definite function.*

**Assumption 4** *The unknown dynamics  $f$  are contained in the RKHS of the chosen kernel  $k(\cdot, \cdot)$ , and an upper bound for its norm is available  $\Gamma \geq \|f\|_{\mathcal{H}}$ .*

The two conditions above are central to the development of the control strategy safety guarantees. Assumption 3 can be satisfied by selecting an appropriate kernel function such as the squared-exponential or the inverse multiquadrics. Assumption 4 encapsulates our knowledge about the complexity of the unknown ground-truth: intuitively, the more kernel basis functions are needed to describe it, the larger the associated norm. The same piece of information is required in the works Koller et al. (2018); Hashimoto et al. (2020) as well as in various other recent papers. In Maddalena et al. (2020), an example is provided on how  $\Gamma$  could be estimated from noiseless samples of the latent function, and how this estimation process is affected by the presence of bounded noise. As shown in the latter work, finite-sample deterministic error bounds exist for KRR models.

**Theorem 1** (Maddalena et al. (2020)) *Let  $K$  be the kernel matrix,  $D$  be the number of data-points,  $\bar{\delta} \in \mathbb{R}_{\geq 0}^D$  the noise bound, and  $\lambda > 0$  be the regularization constant. Under Assumptions 1 to 4, the KRR model  $\hat{f}$  admits the following prediction error bound for any  $z \in \mathcal{Z}$*

$$\left| \hat{f}(z) - f(z) \right| \leq P(z) \sqrt{\Gamma^2 + \Delta - y^\top K^{-1} y} + \bar{\delta}^\top |K^{-1} K_{zz}| + \left| y^\top \left( K + \frac{1}{D\lambda} K K \right)^{-1} K_{zz} \right| \quad (7)$$

where  $f$  is the unknown ground-truth and  $\Delta = \max\{-\delta^\top K^{-1} \delta + 2y^\top K^{-1} \delta\}$  subject to  $|\delta| \leq \bar{\delta}$ .

Notice that the bound above can be easily evaluated, and the only term that is not given in closed-form is the constant  $\Delta$ — which requires solving a box-constrained quadratic program over  $D$  variables. This quantity is independent of the query point  $z$  and compensates for a possible underestimation of the model complexity caused by the noise (Maddalena et al., 2020, Lemma 1). If one wishes not to solve such an optimization problem, then the entire square-root term could be replaced by  $\Gamma$  at the expense of increasing the bound conservativeness. The term  $\bar{\delta}^\top |K^{-1} K_{zz}|$  accounts for the potentially adversarial nature of  $\delta$  and, finally, the last term penalizes the use of high regularization constants  $\lambda$ . For a stable numerical evaluation of (7), a small diagonal *jitter* has to be added to the Gram matrix as customary in the field of Gaussian processes (Bauer et al., 2016).

**Remark 3** When compared to the GP bounds presented in [Srinivas et al. \(2012\)](#), the result given in [Theorem 1](#) does not involve information-theoretic measures such as the maximal information gain. The need of estimating such constant hampers the applicability of the former bounds in practical scenarios (see the discussion in [Lederer et al. \(2019\)](#)). When compared to the results in ([Hashimoto et al., 2020](#), Lemma 2), the inequality (7) tends to give rise to tighter bounds as shown in [Maddalena et al. \(2020\)](#); nevertheless, the latter are more unstable at the extremes of the input space. In order to avoid this effect, data have to ideally fill the ground-truth domain while still being well-separated. In the approximation theory community, this interplay between precision and stability is known as the uncertainty principle ([Wendland, 2004](#)).

#### 4. Kernel Predictive Control

One possible approach to tackling our problem is to build a single-step surrogate model for the latent function and employ uncertainty propagation techniques to perform multi-step ahead predictions. Propagating sets through general non-linear maps is challenging and usually involves several overbounding steps ([Koller et al., 2018](#)). Therefore, we opt for learning various condensed models, one for each of the  $N$  prediction steps. Let  $F_1 : \mathbb{X} \times \mathbb{U} \rightarrow \mathbb{X}$  be the one-step ahead predictor in which each dimension is learned separately by KRR models  $\hat{f}_1, \dots, \hat{f}_{n_x}$

$$(x_0, u_0) \mapsto F_1(x_0, u_0) = \left( \hat{f}_1(x_0, u_0), \dots, \hat{f}_{n_x}(x_0, u_0) \right) \quad (8)$$

Define  $F_2$  as  $(x_0, u_0, u_1) \mapsto F_2(x_0, u_0, u_1) = \left( \hat{f}_1(x_0, u_0, u_1), \dots, \hat{f}_{n_x}(x_0, u_0, u_1) \right)$ , the two-step ahead model, and  $F_3, \dots, F_N$  analogously. Robust *confidence sets* are then built around our nominal predictions. Let  $\mathcal{X}_1 : \mathbb{X} \times \mathbb{U} \rightarrow 2^{\mathbb{X}}$  be a set-valued function defined as the hyper-rectangle

$$(x_0, u_0) \mapsto \mathcal{X}_1(x_0, u_0) = \left( \hat{f}_1(x_0, u_0) \pm \beta_1(x_0, u_0), \dots, \hat{f}_{n_x}(x_0, u_0) \pm \beta_{n_x}(x_0, u_0) \right) \quad (9)$$

where  $\beta(x, u)$  denotes the right-hand side of the inequality (7), and  $a \pm b$  refers to the set  $\{c \mid a - b \leq c \leq a + b\}$ . Similarly, define also the maps  $\mathcal{X}_2, \dots, \mathcal{X}_N$ , which share the same domain respectively with  $F_2, \dots, F_N$ . As a direct consequence of [Theorem 1](#), we have that

$$\begin{aligned} \mathcal{X}_1(x_0, u_0) &\ni f(x_0, u_0) \\ \mathcal{X}_2(x_0, u_0, u_1) &\ni f(f(x_0, u_0), u_1) \\ &\dots \\ \mathcal{X}_N(x_0, u_0, u_1, \dots, u_{N-1}) &\ni f(\dots f(f(x_0, u_0), u_1), \dots, u_{N-1}). \end{aligned} \quad (10)$$

**Remark 4** Training the associated kernel models requires various  $N$ -step experiments to be performed rather than simply having one-step ones. For instance, the model  $F_N$  requires multiple tuples  $(x_0, u_0, \dots, u_{N-1})$  as features and (possibly noisy) measurements of the resulting states  $x_N$  as targets. We highlight that long sequences of linked states, i.e., long experiments, are preferred over various short ones even in classical parametric system identification ([Ljung, 1999](#)).

Let  $x_0$  be a *given* initial condition for the true dynamical system (1). Our Kernel Predictive Control formulation is expressed as the finite-horizon optimal control problem

$$\text{KPC} : \min_{X,U} \sum_{t=0}^{N-1} \ell(x_t, u_t) + \ell_f(x_N) \quad (11a)$$

$$\text{s.t. } x_t = F_t(x_0, u_0, \dots, u_{t-1}), \forall t \quad (11b)$$

$$\mathcal{X}_t(x_0, u_0, \dots, u_{t-1}) \subseteq \mathbb{X}, \forall t \quad (11c)$$

$$\mathcal{X}_N(x_0, u_0, \dots, u_{N-1}) \subseteq \mathbb{X}_{\text{safe}}, \forall t \quad (11d)$$

$$u_t \in \mathbb{U}, \forall t \quad (11e)$$

where  $X = (x_1, \dots, x_N)$ ,  $U = (u_0, \dots, u_{N-1})$  are the decision variables, and  $\ell(x, u)$  and  $\ell_f(x)$  are appropriately designed stage and final costs. In a receding-horizon implementation, KPC is solved recursively and only the first optimal control inputs are applied to the system.

**Proposition 1** *Let the KPC problem (11) be feasible and  $(X^*, U^*)$  be any of its feasible solutions. The sequence of inputs  $U^* = (u_0^*, \dots, u_{N-1}^*)$  drives the true system (1) from  $x_0$  to the safe set  $\mathbb{X}_{\text{safe}}$  while satisfying the constraints at all times, i.e.,  $f(x_0, u_0^*), f(f(x_0, u_0^*), u_1^*), \dots \in \mathbb{X}$ , and  $f(\dots f(f(x_0, u_0^*), u_1^*), \dots, u_{N-1}^*) \in \mathbb{X}_{\text{safe}}$ .*

**Proof:** Follows from the definition of the sets (9), the validity of the deterministic bounds (7), and the imposed constraints (11c) and (11d).

The KPC constraints (11c) and (11d) are set inclusions that need to be reformulated to allow for numerical computations. In what follows, we employ a coordinate transformation and exploit the closed-form solution of the obtained hyper-cube support function. Let  $H_i$  and  $h_i$  be the parameters of the  $i$ th half-space of  $\mathbb{X}$  and consider the condition  $\mathcal{X}_t \subseteq \mathbb{X}$ , where the arguments of  $\mathcal{X}_t$  are omitted to ease notation. This constraint is satisfied if for each one of the half-spaces  $g_i(x) \leq 0$  that describe the polyhedron  $\mathbb{X}$  it holds that

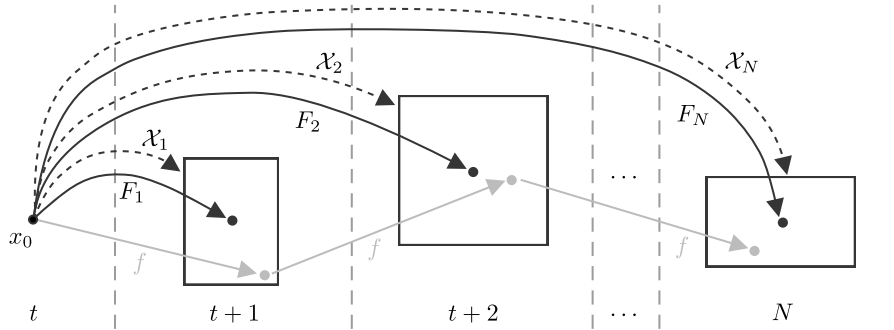


Figure 1: A schematic representation of the outcome of each model. Nominal predictions  $F_t$  (—), confidence sets  $\mathcal{X}_t$  (- -), and the unknown ground-truth  $f$  (· · ·). All functions also depend on the chosen control sequence, which is omitted for clarity.

$$\forall x \in \mathcal{X}_t : g_i(x) \leq 0 \quad (12)$$

$$\Leftrightarrow \max\{g_i(x) \mid x \in \mathcal{X}_t\} \leq 0 \quad (13)$$

$$\Leftrightarrow \max\{H_i^\top x - h_i \mid |x - F_t| \leq \mathcal{B}_t\} \leq 0 \quad (14)$$

$$\Leftrightarrow \max\{H_i^\top (B_t x + F_t) - h_i \mid \|x\|_\infty \leq 1\} \leq 0 \quad (15)$$

$$\Leftrightarrow \|H_i^\top B_t\|_1 + H_i^\top F_t - h_i \leq 0 \quad (16)$$

where  $F_t$  and  $\mathcal{B}_t = (\beta_1, \dots, \beta_{n_x})^\top \in \mathbb{R}^{n_x}$  are the parameters of the confidence set  $\mathcal{X}_t$  (see (9)), and  $B_t = \text{diag}(\beta_1, \dots, \beta_{n_x}) \in \mathbb{R}^{n_x \times n_x}$ . Lastly, the one-norm can be eliminated from (16) by introducing new auxiliary variables and inequality constraints, a standard linear programming procedure.

**Remark 5** The maximization in (14) could also be directly converted into its dual form without the reformulation (15). Although this would not introduce any conservatism, additional decision variables would be created along with *nonlinear equality constraints*, thus significantly increasing the KPC formulation complexity. The approach adopted above is both economic and exact.

After converting the set constraints (11c) and (11d) into the form (16), one obtains

$$\min_{X,U} \sum_{t=0}^{N-1} \ell(x_t, u_t) + \ell_f(x_N) \quad (17a)$$

$$\text{s.t. } x_t = F_t(x_0, u_0, \dots, u_{t-1}), \forall t \quad (17b)$$

$$\|H_i^\top B_t(x_0, u_0, \dots, u_{t-1})\|_1 + H_i^\top F_t(x_0, u_0, \dots, u_{t-1}) - h_i \leq 0, \forall i, \forall t \quad (17c)$$

$$\|Q_i^\top B_N(x_0, u_0, \dots, u_{N-1})\|_1 + Q_i^\top F_N(x_0, u_0, \dots, u_{N-1}) - q_i \leq 0, \forall i, \forall t \quad (17d)$$

$$S u_t \leq s, \forall t \quad (17e)$$

in which  $H_i, Q_i \in \mathbb{R}^{n_x}$ ,  $h_i, q_i \in \mathbb{R}$  for all  $i$ ,  $S \in \mathbb{R}^{n_u \times n_u}$  and  $s \in \mathbb{R}^{n_u}$ . Finally, if a solution to (17) is found, the true system can be steered to  $\mathbb{X}_{\text{safe}}$  in open-loop; however, this feasibility does not guarantee that future iterations of the same problem will also be feasible.

#### 4.1. A Safe Relaxation Strategy

Next we propose a safe relaxation strategy (SRS) that can be used to weaken the optimization problem constraints (17c)-(17d) whenever data from previous KPC iterations are available.

**Proposition 2** Assume that the  $N - 1$  previous consecutive KPC iterations were feasible. Denote by  $(u_{-N+1}, \dots, u_{-1})$  and  $(x_{-N+1}, \dots, x_{-1})$  the closed-loop sequences of past controls and states, and by  $x_0$  the current state. Let  $\text{KPC}^+$  be the KPC optimization problem (11) with set constraints  $\mathcal{X}_1(x_0, u_0) \subseteq \mathbb{X}$ ,  $\mathcal{X}_2(x_0, u_0, u_1) \subseteq \mathbb{X}$ ,  $\dots$ ,  $\mathcal{X}_{N-1}(x_0, u_0, \dots, u_{N-2}) \subseteq \mathbb{X}$  relaxed to

$$\begin{aligned} \mathcal{X}_1(x_0, u_0) \cap \left( \bigcap_{i=2}^N \mathcal{X}_i(x_{-i+1}, u_{-i+1}, \dots, u_{-1}, u_0) \right) &\subseteq \mathbb{X}, \\ \mathcal{X}_2(x_0, u_0, u_1) \cap \left( \bigcap_{i=3}^N \mathcal{X}_i(x_{-i+2}, u_{-i+2}, \dots, u_{-1}, u_0, u_1) \right) &\subseteq \mathbb{X}, \dots \end{aligned} \quad (18)$$

Let  $\text{KPC}^+$  be feasible and  $(X^*, U^*)$  be any of its feasible solutions. Then,  $U^* = (u_0^*, \dots, u_{N-1}^*)$  drives the true system (1) from  $x_0$  to the safe set  $\mathbb{X}_{\text{safe}}$  while satisfying the constraints at all times, i.e.,  $f(x_0, u_0^*), f(f(x_0, u_0^*), u_1^*), \dots \in \mathbb{X}$ , and  $f(\dots f(f(x_0, u_0^*), u_1^*), \dots, u_{N-1}^*) \in \mathbb{X}_{\text{safe}}$ .

**Proof:** From construction, we know that  $\mathcal{X}_t(x_0, u_0^*, \dots, u_{t-1}^*) \ni f(\dots f(f(x_0, u_0^*), \dots), u_{t-1}^*)$  for any  $t$ . If the previous KPC iteration was feasible, then also  $\mathcal{X}_{t+1}(x_{-1}, u_{-1}, u_0^*, \dots, u_{t-1}^*)$  contains the same point, where  $x_{-1}$  and  $u_{-1}$  are past closed-loop data. Considering a total of  $N - 1$  previous consecutive feasible KPC iterations yields a total of  $N$  set conditions for  $f(x_0, u_0^*)$ ,  $N - 1$  set conditions for  $f(x_0, u_0^*, u_1^*), \dots$ , and 1 set condition for  $f(\dots f(f(x_0, u_0^*), u_1^*), \dots, u_{N-1}^*)$ . At any time  $t = 1, \dots, N$ , the true system state is therefore contained in the intersection of the associated sets and, hence, enforcing the relaxed constraints suffices to enforce constraint satisfaction.

The KPC formulation (11) does not guarantee recursive feasibility due to the use of several distinct models. Nevertheless, previous successful iterations can contribute to the feasibility of future KPC problems through the stated SRS. More specifically, the closed-loop data of up to  $N - 1$  past steps<sup>2</sup> can be used to reduce the uncertainty regarding the location of true next states *without updating the KRR nominal models*. We highlight that the last constraint  $x_N \in \mathbb{X}_{\text{safe}}$  is not relaxed by Proposition 2, but remains unchanged.

## 5. Experiments

We illustrate the use of KPC, implemented in a receding-horizon fashion, in two different scenarios. The optimization problems were formulated with the aid of CasADi (Andersson et al., 2019), the Multi-Parametric Toolbox (Herceg et al., 2013), and solved with IPOPT<sup>3</sup>.

**Example 1:** Consider a continuous stirred-tank reactor (CSTR) whose continuous-time dynamics are given by the differential equations

$$\dot{c}_A(t) = u(t)(c_{A0} - c_A(t)) - \rho_1 c_A(t) - \rho_3 c_A(t)^2 \quad (19a)$$

$$\dot{c}_B(t) = -u(t) c_B(t) + \rho_1 c_A(t) - \rho_2 c_B(t)^2 \quad (19b)$$

where  $c_A$  and  $c_B$  denote respectively the concentrations of cyclopentadiene and cyclopentenol, and  $u$  represents the feed inflow of cyclopentadiene. We assume that the reactor temperature is constant and simulate the dynamics with the parameter values:  $\rho_1 = \rho_2 = 4.1 \times 10^{-3} \text{ h}^{-1}$ ,  $\rho_3 = 6.3 \times 10^{-4} \text{ h}^{-1}$ ,  $c_{A0} = 5.1 \text{ mol/l}$ . The constraint sets are  $\mathbb{X} = \{x \in \mathbb{R}^2 \mid (1 \ 0.5)^\top \leq x \leq (3 \ 2)^\top\}$ ,  $\mathbb{U} = \{u \in \mathbb{R} \mid 3 \leq u \leq 35\}$ , and the safe set is the singleton  $\mathbb{X}_{\text{safe}} = \{x_{\text{eq}}\}$ ,  $x_{\text{eq}} = (2.14 \ 1.09)^\top$  with  $u_{\text{eq}} = 14.19$ . The sampling period is 30 seconds and the prediction horizon was chosen to be  $N = 3$ . Three distinct random datasets were collected with 300, 400 and 500 points respectively for the one-step, two-step and three-step ahead predictors. The noise affecting our samples was drawn randomly with the uniform bound  $\bar{\delta} = 1 \times 10^{-3}$ . Squared-exponential kernels were chosen and their hyperparameters were adjusted until good fits were obtained; specifically, the lengthscales were set to larger values when dealing with higher-dimensional feature spaces. The exact regressor norms were calculated and an augmentation factor of 150% was used to obtain estimates  $\Gamma$ . This latter step accounts for the ground-truth complexity in unexplored regions of the space. Finally, standard quadratic stage and terminal costs were employed with positive definite weight matrices.

2. If only  $M < N - 1$  previous iterations were feasible, (18) can be adapted to have less set intersections.

3. Additional details about the simulations are available at <https://github.com/emilioMaddalena/KPC>.



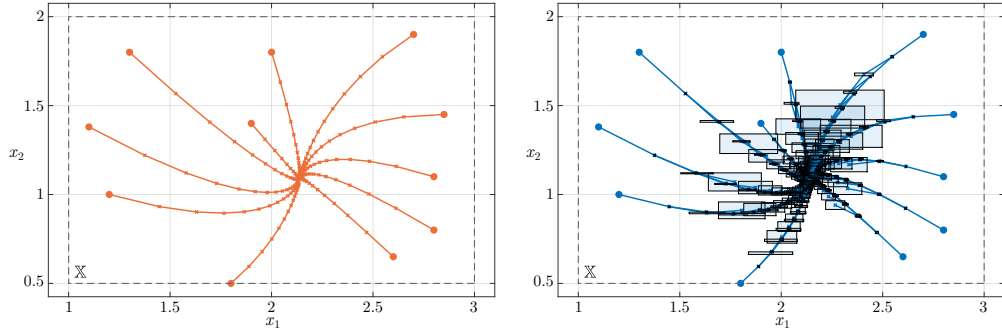


Figure 2: Left: Closed-loop system trajectories starting from different initial conditions. Right: KPC open-loop nominal predictions and uncertainty sets during all time-instants and starting from different initial conditions.

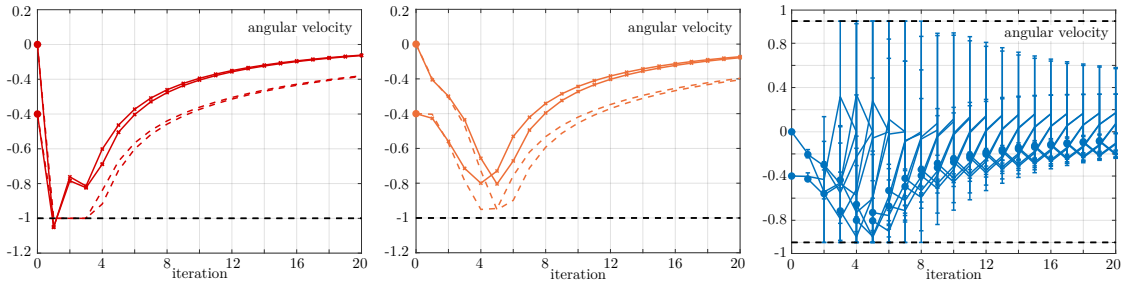


Figure 3: Left: Nominal kernel-MPC predictions (---) and closed-loop trajectory (—). Center: KPC predictions (---) and closed-loop trajectory (—). Right: KPC full open-loop predictions, confidence intervals (—) as error bars, and initial points depicted as circular markers. The nominal kernel-MPC predictions satisfy the constraints, but the real system violates them. KPC leads to a safe operation without constraint violations.

As is customary in practical non-linear optimal control, the terminal constraint was dropped and only a terminal penalty was employed. The system evolution starting from various initial conditions are shown in Figure 2. The closed-loop trajectories (shown on the left) converged to a neighborhood of  $x_S$ , while all predictions and confidence sets (shown on the right) remained inside the feasible set  $\mathbb{X}$  at all time-instants. It is also possible to note how predicting further into the future is more challenging as the lengths of the boxes tended to be larger at the end of the prediction horizon.

**Example 2:** The continuous-time angular dynamics of a pendulum with a rigid rod can be described by  $\dot{x}_1 = x_2$ ,  $\dot{x}_2 = (g/l) \sin(x_1) - (\nu/(ml^2))x_2 + (1/(ml^2))u$ , where  $x_1$  is its angular position,  $x_2$  its angular velocity, and  $u$  the torque applied to it. The parameters are:  $m = 0.15$  the mass of the pendulum,  $l = 0.5$  the length of the rod,  $g = 9.81$  the gravitational constant, and  $\nu = 0.1$  a constant for the friction model. Let the constraints be  $|x_1| \leq 3$ ,  $|x_2| \leq 1$ , and the input be limited to  $|u| \leq 1$ . We select a prediction horizon of  $N = 4$  and collect  $D = 100$  uniformly random data-points for each of the eight regression tasks. The noise was drawn randomly from a uniform distribution with bound  $\bar{\delta} = 0.01$ . Similarly to the previous example, we use a squared-exponential kernel with increasing lengthscales and employ an augmentation factor of 300% on the nominal

predictor norms to estimate the  $\Gamma$  constants. The system was discretized at a frequency of 5 Hz. We compared KPC against *nominal* kernel-MPC, i.e., a certainty equivalence approach where the state-constraints were imposed directly on the nominal predictions (11b). In the latter case, uncertainty was not quantified and the confidence sets were not present. The costs for both formulations were identical positive definite functions of the states and control inputs.

Predictions and the system angular velocity evolution from two different initial conditions are shown in Figure 3. As can be seen from the plots, imposing the state-constraints on the nominal model predictions is not sufficient to guarantee safety as the closed-loop system behavior violates the  $x_2 \geq -1$  restriction. On the other hand, since KPC quantifies and incorporates the associated uncertainty into the optimization problem, the constraints were satisfied. The error bars on the right plot show the all predictions and uncertainty values at each step in the form of error bars. Note that the safety constraints were active at multiple points in time.

## 6. Concluding Remarks

KPC was proposed as a predictive control methodology based on non-parametric kernel models and their associated uncertainty estimates. Its key feature is deterministic constraint satisfaction when a solution to the optimization problem is found. From an approximation theory perspective, future works could study the advantages of employing SVR surrogate models over KRR ones, as well as refining the existing error-bounds, which we believe to be possible. Establishing conditions under which KPC would enjoy additional closed-loop properties such as convergence is deemed as interesting and could guide practical real-world applications of the proposed control scheme.

## Acknowledgments

This work received support from the Swiss National Science Foundation under the Risk Aware Data-Driven Demand Response project (grant number 200021 175627) and CSEM’s Data Program.

## References

- B. D. O. Anderson. Failures of adaptive control theory and their resolution. *Communications in Information & Systems*, 5(1):1–20, 2005.
- J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl. CasADi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1): 1–36, 2019.
- E. Arcari, A. Carron, and M. N. Zeilinger. Meta learning MPC using finite-dimensional Gaussian process approximations. *arXiv preprint arXiv:2008.05984*, 2020.
- M. Bauer, M. van der Wilk, and C. E. Rasmussen. Understanding probabilistic sparse Gaussian process approximations. In *Advances in Neural Information Processing Systems*, pages 1533–1541, 2016.
- T. Beckers, D. Kulić, and S. Hirche. Stable gaussian process based tracking control of euler-lagrange systems. *Automatica*, 103:390–397, 2019.

- M. Bujarbaruah and C. Vallon. Exploiting model sparsity in adaptive MPC: A compressed sensing viewpoint. In *Learning for Dynamics and Control*, pages 137–146, 2020.
- J.-P. Calliess, S. J. Roberts, C. E. Rasmussen, and J. Maciejowski. Lazily adapted constant kinky inference for nonparametric regression and model-reference adaptive control. *Automatica*, 122: 109216, 2020.
- A. Capone, G. Noske, J. Umlauf, T. Beckers, A. Lederer, and S. Hirche. Localized active learning of Gaussian process state space models. In *Learning for Dynamics and Control*, pages 490–499, 2020.
- J. Garcia and F. Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.
- K. Hashimoto, A. Saoud, M. Kishida, T. Ushio, and D. Dimarogonas. Learning-based safe symbolic abstractions for nonlinear control systems. *arXiv preprint arXiv:2004.01879*, 2020.
- M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari. Multi-parametric toolbox 3.0. In *2013 European control conference (ECC)*, pages 502–510. IEEE, 2013.
- L. Hewing, J. Kabzan, and M. N. Zeilinger. Cautious model predictive control using Gaussian process regression. *IEEE Transactions on Control Systems Technology*, 2019.
- L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger. Learning-based model predictive control: Toward safe learning in control. *Annual Review of Control, Robotics, and Autonomous Systems*, 3:269–296, 2020.
- M. Kanagawa, P. Hennig, D. Sejdinovic, and B. K. Sriperumbudur. Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv preprint arXiv:1807.02582*, 2018.
- G. S. Kimeldorf and G. Wahba. A correspondence between bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41(2):495–502, 1970.
- T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause. Learning-based model predictive control for safe exploration. In *IEEE Conference on Decision and Control (CDC)*, pages 6059–6066. IEEE, 2018.
- A. Lederer, J. Umlauf, and S. Hirche. Uniform error bounds for gaussian process regression with application to safe control. In *Advances in Neural Information Processing Systems*, pages 659–669, 2019.
- Y. Lian and C. N. Jones. On Gaussian process based koopman operators. *IFAC-PapersOnLine*, 53(2):449–455, 2020.
- L. Ljung. System identification: theory for the user. *PTR Prentice Hall, Upper Saddle River, NJ*, pages 1–14, 1999.
- M. Lorenzen, M. Cannon, and F. Allgöwer. Robust MPC with recursive model update. *Automatica*, 103:461–471, 2019.

- E. T. Maddalena and C. N. Jones. NSM converges to a k-NN regressor under loose Lipschitz estimates. *IEEE Control Systems Letters*, 4(4):880–885, 2020.
- E. T. Maddalena, P. Scharnhorst, and C. N. Jones. Deterministic error bounds for kernel-based learning techniques under bounded noise. *arXiv preprint arXiv:2008.04005*, 2020.
- J. M. Manzano, D. Limon, D. M. de la Peña, and J.-P. Calliess. Robust learning-based MPC for nonlinear constrained systems. *Automatica*, 117:108948, 2020.
- J. Matschek and R. Findeisen. Learning supported model predictive control for tracking of periodic references. In *Learning for Dynamics and Control*, pages 1–10, 2020.
- C. A. Micchelli, Y. Xu, and H. Zhang. Universal kernels. *Journal of Machine Learning Research*, 7(Dec):2651–2667, 2006.
- M. Milanese and C. Novara. Set membership identification of nonlinear systems. *Automatica*, 40(6):957–975, 2004.
- B. Schölkopf, A. J. Smola, and F. Bach. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. the MIT Press, 2018.
- H. A. Shukla, T. de A. Ferreira, T. Faulwasser, D. Bonvin, and C. N. Jones. Convergence certificate for stochastic derivative-free trust-region methods based on Gaussian processes. *arXiv preprint arXiv:2010.01120*, 2020.
- 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.
- M. Tanaskovic, L. Fagiano, and V. Gligorovski. Adaptive model predictive control for linear time varying MIMO systems. *Automatica*, 105:237–245, 2019.
- J. Umlauft and S. Hirche. Learning stochastically stable Gaussian process state-space models. *IFAC Journal of Systems and Control*, 12:1–15, 2020.
- K. P. Wabersich and M. N. Zeilinger. Performance and safety of Bayesian model predictive control: Scalable model-based RL with guarantees. *arXiv preprint arXiv:2006.03483*, 2020.
- H. Wendland. *Scattered data approximation*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2004. ISBN 9781139456654.
- C. K. I. Williams and C. E. Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.