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

Online learning robust MPC: an exploration-exploitation approach ⋆

J.M. Manzano[1], J. Calliess[2], D. Muñoz de la Peña[1], D. Limon[1]

Abstract: This paper presents a predictive controller whose model is based on input-output data of the nonlinear system to be controlled. It uses a Lipschitz interpolation technique in which new data may be included in the database in real time, so the controller improves the system model online. An exploration and exploitation policy is proposed, allowing the controller to robustly and cautiously steer the system to the best reachable reference, even if the model lacks data in such region. The conditions needed to ensure recursive feasibility in the presence of output and input constraints and in spite of the uncertainties are given. The results are illustrated in a simulated case study.

Keywords: Learning control; Nonlinear control; Target tracking; Robust stability; Predictive control; Sampled-data systems; Output feedback

1. INTRODUCTION

In the light of growing successes of machine learning algorithms, data-based control techniques have become increasingly popular in the control community in recent years. Models based on data have proven to be able to emulate the system behaviour well (Aswani et al., 2013), and thus they come especially handy when accurate first-principles models of the system are not available. In these cases, a machine learning technique is used to learn and to predict the evolution of the plant. Such techniques may handle data in a deterministic or stochastic way, e.g., Lipschitz interpolation (Canale et al., 2014) for the former and Gaussian processes (Akametalu et al., 2014; Berkenkamp and Schoellig, 2015) for the latter. These approaches have been applied in different model predictive controllers (MPC), as reviewed in Hewing et al. (2019).

When controlling a system, taking into account new information from the operation of such system seems to be the order of the day within the control research community. It also seems intuitive that the current trajectories of the closed-loop system could be considered in order to improve the controller. With respect to these approaches, one may come across terms such as *online* or *learning*. While being a popular topic in reinforcement learning (RL) (Mnih et al., 2015; Lillicrap et al., 2015) and adaptive control (Adetola et al., 2009; Tanaskovic et al., 2019) , the matter of online learning still seems to be somewhat under-researched in data-based model-predictive control (Di Cairano et al., 2013; Ostafew et al., 2014).

These many papers could be classified according to the use they make of new data, like improving an initial feasible solution in an optimization problem (Schwenkel et al., 2018), designing safety filters (Wabersich and Zeilinger, 2018), or improving a whole model of the system (Hewing et al., 2018). They could also be classified regarding this model of the system, ranging from linear models (Lorenzen et al., 2019) to data-based models.

This paper is based on the knowledge of a data set of past inputs and outputs trajectories of the plant, and uses a Lipschitz interpolation technique for the model, which has interesting properties suitable for data-based MPCs. If the data set is dense enough to guarantee a bound on the prediction error sufficiently small, robust MPCs can be designed (Manzano et al., 2019b, 2020). Besides, if the prediction model is updated online, the performance of the controlled system may be enhanced (Limon et al., 2017).

In this paper, a learning MPC is proposed to deal with the case of low-dense data sets. To this end, an exploration policy is developed, which yields a cautious estimation of the system by bounding the prediction error to the robustness margin permitted by the controller. An exploitation methodology regulates the inclusion of new data points, preventing the data set from growing excessively.

Based on these techniques, a predictive controller for tracking (Limon et al., 2018) is proposed. It will be able to robustly steer the system to changing references, safely exploring regions which lack data, while satisfying hard constraints in both inputs and outputs and maintaining stability and recursive feasibility.

Notation: $i \in \mathbb{I}_a^b$ stands for the set of integers $i = a, \dots, b$. The Minkowski sum of two sets A, B is denoted $A \oplus B$ and the Pontryagin difference $A \ominus B$. Given two column vectors v, w, (v, w) stands for $[v^T, w^T]^T$. A ball of radius r is denoted $\mathcal{B}(r) = \{x : \|x\| \leq r\}$. A function $\alpha(\cdot)$ is a $\mathcal{K}$-function if it is strictly increasing a $\alpha(0) = 0$. Besides, if it is not upper bounded, it is called a $\mathcal{K}_\infty$-function.

⋆ The authors would like to thank the funds received by the MINECO and Feder Funds under contract DPI2016-76493-C3-1-R and the VI-PPIT of the University of Seville.

[1]Departamento de Ingeniería de Sistemas y Automática, Universidad de Sevilla, Spain. {manzano,dmunoz,dlm}@us.es

[2]Oxford-Man Institute of Quantitative Finance. University of Oxford, UK. jan-peter.calliess@oxford-man.ox.ac.uk

2. PROBLEM SETTING

The objective of the paper is to control a discrete time system whose model is unknown. Only inputs $u(k) \in \mathbb{R}^{n_u}$ and outputs $y(k) \in \mathbb{R}^{n_y}$ measurements are available. It is assumed that such system can be described by a nonlinear auto-regressive exogenous model (NARX) (Levin and Narendra, 1997), with the form

$$y(k+1) = f(x(k), u(k)) + e(k), \tag{1}$$

where the state can be represented as

$$\begin{aligned} x(k) = (&y(k), \dots, y(k-n_a), \\ &u(k-1), \dots, u(k-n_b)), \end{aligned} \tag{2}$$

for some memory horizons $n_a, n_b \in \mathbb{N}$ [2]. The noise is assumed to be bounded, such that $|e(k)| \leq \bar{\mathfrak{e}}$. The inputs are constrained by $u(k) \in \mathcal{U}$ and the outputs by $y(k) \in \mathcal{Y}$, or, in other words, $(y, u) \in \mathcal{Z}$.

As in Manzano et al. (2019b), the terms $x(k)$ and $u(k)$ are aggregated into a joint variable, called *regressor*,

$$w(k) = (x(k), u(k)) \in \mathbb{R}^{n_w}, \tag{3}$$

with $n_w = n_y(n_a + 1) + n_u(n_b + 1)$.

Given a data set of previous inputs and outputs

$$\mathcal{D}_{\text{raw}} = \{(u_i, y_i) : i = 1, \dots, N_{\text{raw}}\},$$

it is aggregated as stated before, obtaining

$$\mathcal{D} = \{(w_i, \tilde{f}(w_i)) : i = 1, \dots, N_{\mathcal{D}}\}, \tag{4}$$

where $\tilde{f}$ stands for the noisy observation of f. The data set containing only reggresors is accordingly named $\mathcal{W}_{\mathcal{D}}$.

For a given reggresor (probably not included in $\mathcal{W}_{\mathcal{D}}$) it is possible to predict its output using a function $\hat{\mathfrak{f}}$, obtained with a machine learning technique. That is,

$$\hat{y}(k+1) = \hat{\mathfrak{f}}(w(k); \theta, \mathcal{D}), \tag{5}$$

where $\theta \in \mathbb{R}^{n_\theta}$ stands for the (hper-)parameters needed by the chosen method.

In particular, this paper makes use of a class of Lipschitz interpolation techniques (Beliakov, 2006) known as kinky inference (KI) (Calliess, 2014). To this end, the ground-truth function f is required to be Lipschitz continuous, with Lipschitz constant L^*, such that

$$\|f(w_1) - f(w_2)\| \leq L^* \|w_1 - w_2\|.$$

In general, this constant L^* is unknown. However, given the data set $\mathcal{D}$, one can estimate a lower bound as follows (Calliess, 2016):

$$L_{\mathcal{D}} = \max_{(w \neq w') \in \mathcal{W}_{\mathcal{D}}} \frac{\|\hat{f}(w) - \hat{f}(w')\| - \eta}{\|w - w'\|}, \tag{6}$$

where the regularization term is set to $\eta = 2\bar{\mathfrak{e}}$.

Using this constant, KI computes each component $j \in \mathbb{I}_1^{n_y}$ of the future output as:

$$\begin{aligned} \hat{\mathfrak{f}}_j(q; L_{\mathcal{D}}, \mathcal{D}) = &\frac{1}{2} \min_{i \in \mathbb{I}_1^{N_{\mathcal{D}}}} \left(\tilde{f}_{i,j} + L_{\mathcal{D}} \|q - w_i\| \right) \\ &+ \frac{1}{2} \max_{i \in \mathbb{I}_1^{N_{\mathcal{D}}}} \left(\tilde{f}_{i,j} - L_{\mathcal{D}} \|q - w_i\| \right). \end{aligned} \tag{7}$$

[2] For an analysis on how to estimate the horizons n_a, n_b, please refer to Manzano et al. (2019a).

This inference method has proven capable of conforming a valid model to be used for prediction in a MPC framework with robust stability guarantees (Manzano et al., 2020).

The estimation error between the predictor and the real function is denoted $d(k)$, and its maximum value μ:

$$\|f(x(k), u(k)) - \hat{\mathfrak{f}}(x(k), u(k))\| = d(k) \leq \mu. \tag{8}$$

Note that the data-based model (7) can be extended to state space provided that

$$\hat{x}(j+1|k) = \hat{F}(\hat{x}(j|k), u(k+j); L_{\mathcal{D}}, \mathcal{D}) \tag{9}$$

$$\hat{y}(j|k) = M\hat{x}(j|k), \tag{10}$$

where $M = [I_{n_y}, 0, \dots, 0]$ and

$$\begin{aligned} \hat{F}(\hat{x}(j|k), u(k+j)) = (&\hat{\mathfrak{f}}(\hat{x}(j|k), u(k+j)), \\ &\hat{y}(j|k), \cdots, y(k), \cdots, \\ &y(k+j-n_a+1), u(k+j), \\ &\cdots, u(k+j-n_b+1)). \end{aligned}$$

3. EXPLORATION-EXPLOITATION METHODOLOGY

When designing online learning methods, a trade-off between *exploration* and *exploitation* comes up. We use the term exploration to measure how far from the known workspace the system is allowed to move. By exploitation we address the fact that obtaining a large data set may not be the best strategy to follow, computationally speaking. The kinky inference technique presented in the previous section is especially suitable for online learning:

3.1 Exploration

The objective is the design of predictive controllers able to control the system in regions with low data density. In such areas the prediction error increases rapidly, probably exceeding the robustness bound that the controller can afford.

This problem can be tamed considering an exploration technique in which the control strategy forces the system to stay *close* to a safe region, where we have enough information to guarantee a worst case upper bound of the prediction error. This *safe region* is defined as

$$\mathcal{W}_r = \left\{ w : \min(\|w - w_i\|) \leq \tau_r, \forall i \in \mathbb{I}_1^{N_{\mathcal{D}}} \right\}, \tag{11}$$

for certain threshold $\tau_r \geq 0$.

The following property [3] allows the method to relate the exploration distance τ_r with the estimation error bound.

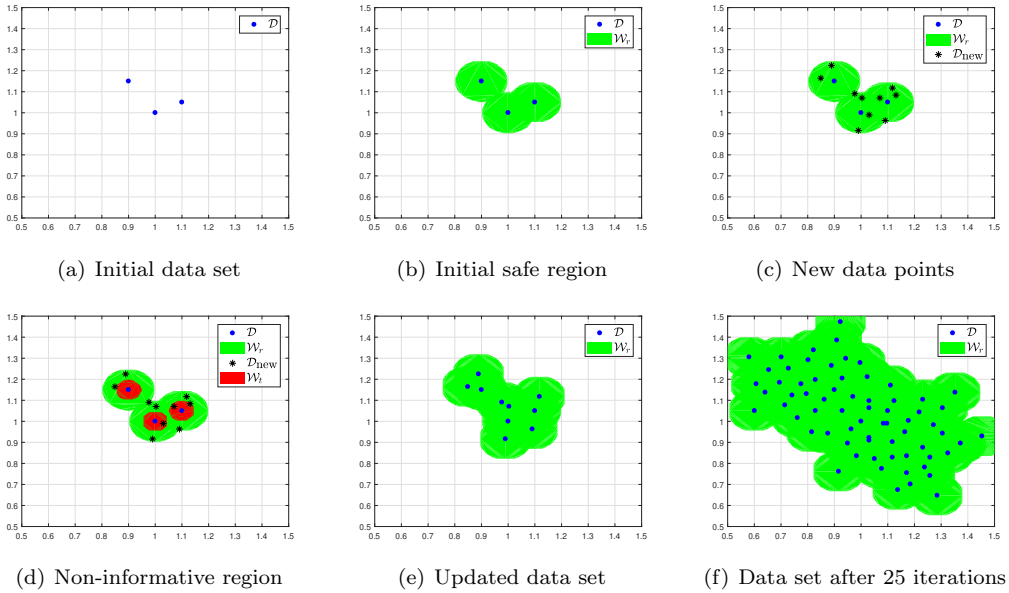
Property 1. *The prediction error μ is bounded by*

$$\mu = L^* \tau_r + 2\bar{\mathfrak{e}}. \tag{12}$$

Note that this bound is based on the true Lipschitz constant L^*, which in general is unknown. In this work we assume that this constant is known, equal to the estimated constant $L_{\mathcal{D}}$.

Assumption 1. $L_{\mathcal{D}} = L^*$.

[3] The proof is omitted in this version, due to the limited number of pages



(a) Initial data set (b) Initial safe region (c) New data points

(d) Non-informative region (e) Updated data set (f) Data set after 25 iterations

Fig. 1. Exploration-exploitation algorithm

The veracity of Assumption 1 depends on the density of the data set, and it conditions the validity of the results presented in this paper. Previous works on the estimation of the Lipschitz constant (Calliess, 2015) provide a Pareto probability distribution on the Bayesian estimation of L^*. Besides, if the probability of an underestimation is bounded by ρ, i.e., $\Pr(L^* > L_{\mathcal{D}}) \leq \rho$, the applicability of this paper is extended to a confidence level $(1-\rho)$.

3.2 Exploitation

It has been proven that the prediction error $d(k)$ vanishes when the density of the data set becomes infinite (Calliess, 2016). In practice, adding data points to the model increases computation times, and hence, it may not be the ideal procedure to include every new data point observed. Instead, in this work we propose to implement an exploitation policy, adding only *informative* data points, that is, those that are not close to data points already seen.

We characterize the term *close* by another threshold of the distance, such that a new data point q is not informative if it belongs to the *well-known* region, defined as

$$\mathcal{W}_t = \left\{ w : \min(\|w - w_i\|) \leq \tau_t,\ \forall i \in \mathbb{I}_1^{N_{\mathcal{D}}} \right\}, \tag{13}$$

for certain threshold $0 \leq \tau_t \leq \tau_r$.

This threshold hyperparameter has to be appropriately chosen, according to the information added by the inclusion of q in $\mathcal{D}$. Kingravi (2014) studied the dependence of new data points w.r.t. the data stored in $\mathcal{D}$, in the context of Gaussian processes for learning-based control. A procedure to prune uninformative sample points given a Lipschitz constant estimation is given in Calliess (2014).

An example of the exploration-exploitation algorithm for a two-dimensional input space is shown in Figure 1. In this figure, an initial data set is considered with $N_{\mathcal{D}} = 3$. New data points are drown randomly within the safe region $\mathcal{W}_r$, but only added if they do not belong to $\mathcal{W}_t$, with $\tau_r = 0.1$ and $\tau_t = 0.05$. Besides, the 2-norm was chosen as metric: $\|q - w\|_2$. After 25 iterations $N_{\mathcal{D}} = 68$.

The data set is updated every time step, denoting $\mathcal{D}(k)$ the data set at time instant k, and $\mathcal{D}(0)$ the initial one. Note that the safe and the well-known regions are also time-dependent (i.e., $\mathcal{W}_r(k), \mathcal{W}_t(k)$). This update policy is such that

$$\mathcal{D}(k+1) = \begin{cases} \mathcal{D}(k) & \text{if } w(k) \in \mathcal{W}_t(k) \\ \mathcal{D}(k) \cup (y(k+1), w(k)) & \text{if } w(k) \notin \mathcal{W}_t(k). \end{cases} \tag{14}$$

The same occurs with the estimation of the Lipschitz constant. The recalculation of $L_{\mathcal{D}}(k)$ is done recursively (Calliess, 2016) as per (6). It can be proven that this estimation tends to the real L^* when the data set becomes infinitely dense. Computationally, this recursion is linear w.r.t. the cardinality of the data set, $\mathcal{O}(N_{\mathcal{D}})$, in contrast to other existing methods (such as Gaussian processes, which are quadratic, $\mathcal{O}(N_{\mathcal{D}}^2)$).

Notice that from the computation point of view, the exploration-exploitation algorithms barely increase the calculation times. It is not necessary to calculate a closed-form of the sets $\mathcal{W}_r(k)$ and $\mathcal{W}_t(k)$. Instead, it is only necessary to check whether a given query point q belongs to them. This is carried out evaluating the minimum distance to the data set $\mathcal{W}_D$; that is,

$$q \in \begin{cases} \mathcal{W}_t \\ \mathcal{W}_r \end{cases} \text{ iff } \min \|q - w_i\| \leq \begin{cases} \tau_t \\ \tau_r \end{cases}, \ \forall i \in \mathbb{I}_1^{N_{\mathcal{D}}}.$$

It is also important to remark that these distances $\|q - w_i\|$ are used to make predictions for that regressor (cf. eq. (7)), so they are already obtained in the prediction step.

4. ONLINE-LEARNING CONTROLLER

In this section, a predictive controller that makes use of the data-based prediction model (5) and the exploration-exploitation approach of Section 3 is presented. In order to be able to follow references (possibly outside of the

initial data set), we propose to use a robust MPC for tracking (Limon et al., 2018) that takes into account explicitly at each time step the current safe region to guarantee a given uncertainty bound on the predictions. MPC for tracking is designed to guarantee stability in the presence of sudden reference changes, even if they are not reachable, which may be the case in the exploration scenario.

To this end, the MPC optimization problem considers an artificial reference (u_s, y_s) as additional decision variables. The deviation of the system to this reference is penalised along the prediction horizon, by means of a stage cost of the form $\ell(y - y_s, u - u_s)$. A term $V_O(y_s - y_t)$ is added to the cost function, in order to penalise the deviation of the artificial reference to the true reference (u_t, y_t).

Therefore, we propose the following controller, capable of exploring unseen regions, by forcing the system to stay in the explored area $\mathcal{W}_r$. Combined with the exploitation algorithm of adding data points only if $q \notin \mathcal{W}_t$, one can step by step move onto unexplored areas, while maintaining a prediction error bound suitable to ensure robust stability. The resulting optimization problem P_N is:

$$
\begin{aligned}
\min_{\mathbf{u}, y_s, u_s} \quad & V_N(x(k), \mathbf{u}, u_s, y_s; y_t) \\
& = \sum_{i=0}^{N-1} \ell(\hat{y}(i|k), u(i); y_s, u_s) \\
& + V_O(y_s - y_t) && \text{(15a)} \\
\text{s.t.} \quad & \hat{x}(0|k) = x(k) && \text{(15b)} \\
& \hat{x}(j+1|k) = \hat{F}(\hat{x}(j|k), u(j)), \quad j \in \mathbb{I}_0^{N-1} && \text{(15c)} \\
& \hat{y}(j|k) = M\hat{x}(j|k) && \text{(15d)} \\
& u(j) \in \mathcal{U} && \text{(15e)} \\
& \hat{y}(j|k) \in \mathcal{Y}_j, \quad j \in \mathbb{I}_1^N && \text{(15f)} \\
& u_s \in \lambda\mathcal{U} && \text{(15g)} \\
& y_s = \hat{\mathfrak{f}}(x_s, u_s) && \text{(15h)} \\
& y_s \in \mathcal{Y}_N && \text{(15i)} \\
& \hat{y}(N|k) = y_s && \text{(15j)} \\
& \hat{w}(j|k) \in \mathcal{W}_r(k), \quad j \in \mathbb{I}_1^N && \text{(15k)} \\
& w_s \in \mathcal{W}_r(k), && \text{(15}\ell\text{)}
\end{aligned}
$$

where $x_s = (y_s, \ldots, y_s, u_s, \ldots, u_s)$, $w_s = (x_s, u_s)$, $\hat{x}(j|k)$ stands for the predicted state at time step j given the measurements at time step k, $\hat{w}(j|k)$ is defined as in (3) and $\lambda \lessapprox 1$ is a design parameter.

Note that a terminal equality constraint is included. Besides, a set of tightened constraints $\mathcal{Y}_j$ is considered (Limon et al., 2002). They are defined as

$$\mathcal{Y}_j = \mathcal{Y} \ominus \mathcal{B}(d_j(\mu)), \tag{16}$$

where d_j is a function of μ that can be calculated as in Manzano et al. (2020).

The set of tightened constraints counteracts the effect of the error between the real plant and the data-based model(which is bounded by μ) on the constraints. Hence, this set must not be empty for any $j \in \mathbb{I}_1^N$. This is stated as an assumption in Manzano et al. (2020), conditioning the feasibility of the controller.

In the proposed approach, the bound in the prediction error within the safe regions is a design parameter (see Property 1). This implies that in general, τ_r is chosen so that $\mathcal{Y}_N$ is not empty, which is an important property from the implementation point of view. If the error bound for the whole state space were considered, it could be too large to obtain non empty tightened constraints.

Given the procedure to calculate such sets (Manzano et al., 2020), and the definition of μ in (12), the maximum admissible value of the exploration radius $\tau_r^{\max}$ can be explicitly obtained, provided that Assumption 1 holds.

4.1 Stability analysis

The ingredients of the optimization problem are required to satisfy the following assumption:

Assumption 2. *(1) The stage cost function $\ell(y, u; y_s, u_s)$ is a positive definite function and $\ell(y, u) \leq \alpha_y(\|y - y_s\|) + \alpha_u(\|u - u_s\|)$, for two $\mathcal{K}$-functions α_y, α_u.*

(2) The offset cost function $V_O(y_s - y_t)$ is a subdifferentiable convex positive definite function such that the best reachable reference

$$y_s^0 = \arg\min_{y_s \in \mathcal{Y}_N} V_O(y_s - y_t)$$

is unique; and

$$V_O(y_s - y_t) - V_O(y_s^0 - y_t) \geq \alpha_O(|y_s - y_s^0|),$$

for a given $\mathcal{K}_\infty$-function α_O.

Define $\mathcal{Y}_s$ as a convex set of reachable equilibrium points,

$$\mathcal{Y}_s \subseteq \{y : \exists u_s \in \lambda\mathcal{U} : \hat{\mathfrak{f}}(x_s, u_s) = y_s\}, \tag{17}$$

then the following assumptions are needed to derive the recursive feasibility of the optimization problem.

Assumption 3. $\mathcal{Y}_{N-1} \subseteq \mathcal{Y}_s$.

Assumption 4. *For all x such that $y \in \mathcal{Y}_{N-1}$ there exists a continuous $u_F = \kappa_F(y) \in \mathcal{U}$ such that $\hat{\mathfrak{f}}(x, u_F) \in \mathcal{Y}_N$.*

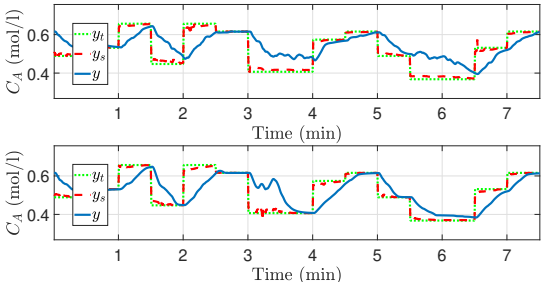
Theorem 1. *Suppose that Assumptions 1-4 hold for the optimization problem P_N. Let $\kappa_N(x)$ be the control law derived from the solution of P_N applied using a receding horizon policy. Then, for any $x(0) \in \mathcal{Z}$, the system controlled by the control law $u(k) = \kappa_N(x(k))$ is recursively feasible, stable, and the constraints are always satisfied, i.e. $u(k) \in \mathcal{U}$, $y(k) \in \mathcal{Y}$, $\forall k$.* [3]

Corollary 1 (Convergence)**.** *In case that the prediction error of the model ($\mu(k)$) tends to 0, the system converges to the best reachable reference y_s^0.* [3]

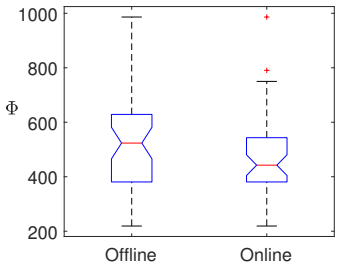
Remark 1. *Note that the online algorithm presented here decreases the prediction error while operating the system. However, even in the absence of the exploitation policy (i.e. $\tau_t = 0$) and infinitely dense data sets, the maximum prediction error μ vanishes up to the level of noise (Calliess, 2016).*

5. CASE STUDY

The continuously stirred tank reactor (CSTR) presented in Seborg et al. (1989) is considered. The manipulable input is the reference temperature of the coolant T_r (K). The measurable output is the concentration of the reactant C_A (mol/l), which evolves according to the set of differential equations given in Manzano et al. (2019b) (as well as the parameters). It is assumed that the concentration measurements have an error of 2% of the signal, which



(a) Output for the offline MPC (above) and online MPC (below)



(b) Performance indexes for 100 simulations

Fig. 2. Comparison between the offline and online MPCs

is generated randomly using an uniform distribution. The constraints in the input are $300\,\mathrm{K} \leq T_r \leq 400\,\mathrm{K}$, and in the output $0 \leq C_A \leq 0.88\,\mathrm{mol/l}$.

5.1 Online learning

We first consider a case in which, at the beginning of the simulation, very few data points are known, just 300 corresponding to some equilibrium points of the system. The regressors are constructed for such data set, with $n_a = 2$ and $n_b = 0$. The estimation of $L_{\mathcal{D}}(0)$ is 1.62.

To motivate the online inclusion of data points while operating the system, we apply the proposed MPC (15), with 15 references varying randomly among $340\,\mathrm{K} \leq T_r \leq 360\,\mathrm{K}$, each of them maintained for 20 s. The controller's parameters are set to $N = 3$, $Q = 10$, $R = 1$ and $O = 100$. The exploration-exploitation is not considered in this example, i.e., $\tau_r = \infty$ and $\tau_t \gtrapprox 0$.

We compare two controllers subject to the same random noise, with and without the online updating policy, for 100 simulations. The results are represented in Figure 2. The behaviour is measured by the performance index, defined as

$$\Phi = \sum_{i=1}^{t_{\mathrm{sim}}} \ell(y(i), u(i), y_s(i), u_s(i)) + V_O(y_s(i) - y_t(i)). \quad (18)$$

The results show that the proposed controller is able to follow the reference better than a controller that does not update the data set, incurring into a smaller cost.

5.2 Exploring

Consider that this same CSTR has historically been operated within the region comprised by $335 \leq T_r \leq 370\,\mathrm{K}$.

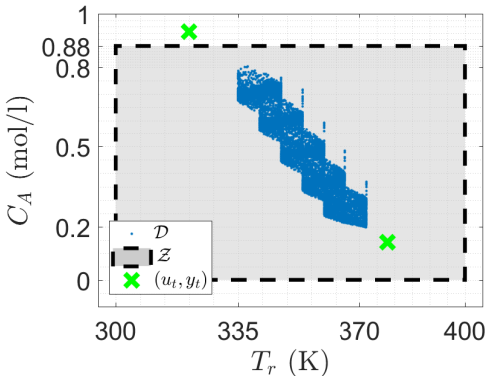


Fig. 3. Input-output space showing the constraints $\mathcal{Z}$, the initial data set $\mathcal{D}(0)$, and the references (u_t, y_t).

Therefore, a large data set within this region is available, as shown in Figure 3. Picture that the owners consider operating the tank in other temperatures, where nothing is known of how the system behaves.

The initial data set yields $L_{\mathcal{D}}(0) = 1.62$. Assuming this as the true Lipschitz constant, and setting $N = 2$, the maximum exploration radius such that $\mathcal{Y}_N$ is not empty is $\tau_r^{\max} = 0.10$ [4], provided that $\bar{\epsilon} = 0.02\,\mathrm{mol/l}$. The proposed controller (15) is applied, with two piece-wise constant references: $y_t = 0.14\,\mathrm{mol/l}$ and $0.93\,\mathrm{mol/l}$, each of them lasting 2 min. Note that both references are in the unexplored area (Fig. 3). Besides, the second one is not even admissible. The radius for exploitation is set to $\tau_t = 0.002$, and for exploration $\tau_r = 0.6\tau_r^{\max}$, to mitigate the possible effect of the underestimation of L^*.

The result of the simulation is shown in Figure 4. Note how the data set is increased with the points visited throughout the operation. Observe also the trajectory of the optimal artificial reference and its convergence to the best reachable steady state. The closed-loop system reaches the real reference, even if it was not reachable in the beginning. In the second part, the robust MPC prevents the closed-loop system from violating the constraints, by means of the set of tightened constraints, while steering the system to the closest reachable state.

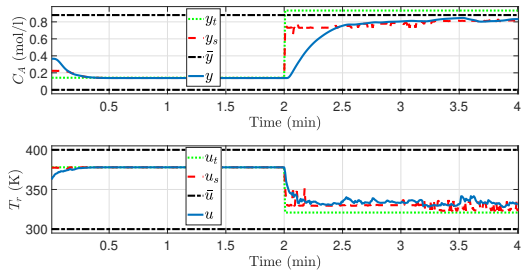
Without the exploration-exploitation algorithm presented in this paper, the closed-loop system would fail to converge to the given reference. On the other hand, if no restriction is added on how far from known data points the system can go, the prediction error increases immensely, being unable to properly forecast the evolution of the plant and therefore to fulfill the constraints.

REFERENCES

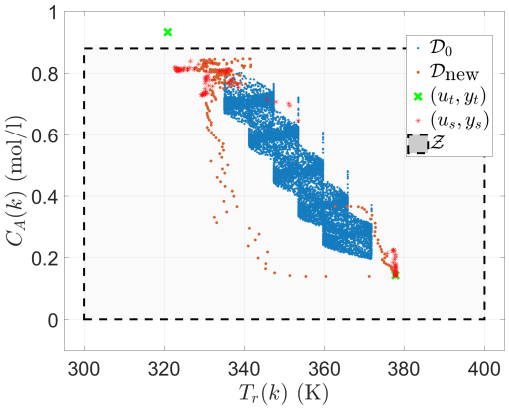
Adetola, V., DeHaan, D., and Guay, M. (2009). Adaptive model predictive control for constrained nonlinear systems. *Systems & Control Letters*, 58(5), 320–326.

Akametalu, A.K., Fisac, J.F., Gillula, J.H., Kaynama, S., Zeilinger, M.N., and Tomlin, C.J. (2014). Reachability-based safe learning with Gaussian processes. In *53rd IEEE Conference on Decision and Control*, 1424–1431. IEEE.

[4] Recall that every signal is scaled to range $0 - 1$.



(a) Closed-loop system



(b) Data points in the input-output space

Fig. 4. Online learning MPC

Aswani, A., Gonzalez, H., Sastry, S.S., and Tomlin, C. (2013). Provably safe and robust learning-based model predictive control. *Automatica*, 49(5), 1216–1226.

Beliakov, G. (2006). Interpolation of Lipschitz functions. *Journal of computational and applied mathematics*, 196(1), 20–44.

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

Calliess, J.P. (2016). Lazily adapted constant kinky inference for nonparametric regression and model-reference adaptive control. *arXiv preprint arXiv:1701.00178*.

Calliess, J.P. (2014). *Conservative decision-making and inference in uncertain dynamical systems*. Ph.D. thesis, University of Oxford.

Calliess, J.P. (2015). Bayesian Lipschitz constant estimation and quadrature.

Canale, M., Fagiano, L., and Signorile, M.C. (2014). Nonlinear model predictive control from data: a set membership approach. *International Journal of Robust and Nonlinear Control*, 24(1), 123–139.

Di Cairano, S., Bernardini, D., Bemporad, A., and Kolmanovsky, I.V. (2013). Stochastic MPC with learning for driver-predictive vehicle control and its application to HEV energy management. *IEEE Transactions on Control Systems Technology*, 22(3), 1018–1031.

Hewing, L., Liniger, A., and Zeilinger, M.N. (2018). Cautious NMPC with Gaussian process dynamics for autonomous miniature race cars. In *2018 European Control Conference (ECC)*, 1341–1348. IEEE.

Hewing, L., Wabersich, K.P., Menner, M., and Zeilinger, M.N. (2019). Learning-based model predictive control: Toward safe learning in control. *Annual Review of Control, Robotics, and Autonomous Systems*, 3.

Kingravi, H. (2014). *Reduced-set models for improving the training and execution speed of kernel methods*. Ph.D. thesis, Georgia Institute of Technology.

Levin, A. and Narendra, K. (1997). Identification of nonlinear dynamical systems using neural networks. In *Neural systems for control*, 129–160. Elsevier.

Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Limon, D., Alamo, T., and Camacho, E. (2002). Input-to-state stable MPC for constrained discrete-time nonlinear systems with bounded additive uncertainties. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 4, 4619–4624. IEEE.

Limon, D., Calliess, J., and Maciejowski, J. (2017). Learning-based nonlinear model predictive control. *IFAC-PapersOnLine*, 50(1), 7769–7776.

Limon, D., Ferramosca, A., Alvarado, I., and Alamo, T. (2018). Nonlinear mpc for tracking piece-wise constant reference signals. *IEEE Transactions on Automatic Control*, 63(11), 3735–3750.

Lorenzen, M., Cannon, M., and Allgöwer, F. (2019). Robust MPC with recursive model update. *Automatica*, 103, 461–471.

Manzano, J., Nadales, J., de la Peña, D.M., and Limon, D. (2019a). Oracle-based economic predictive control. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, 4246–4251. IEEE.

Manzano, J.M., Limon, D., Muñoz de la Peña, D., and Calliess, J.P. (2019b). Output feedback MPC based on smoothed projected kinky inference. *IET Control Theory & Applications*, 13(6), 795–805.

Manzano, J.M., Limon, D., Muñoz de la Peña, D., and Calliess, J.P. (2020). Robust learning-based MPC for nonlinear constrained systems. *Automatica*, 117, 108948.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540), 529.

Ostafew, C.J., Schoellig, A.P., and Barfoot, T.D. (2014). Learning-based nonlinear model predictive control to improve vision-based mobile robot path-tracking in challenging outdoor environments. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 4029–4036. IEEE.

Schwenkel, L., Gharbi, M., Trimpe, S., and Ebenbauer, C. (2018). Online learning with stability guarantees: A memory-based real-time model predictive controller. *arXiv preprint arXiv:1812.09582*.

Seborg, D.E., Edgar, T.F., and Mellichamp, D.A. (1989). *Process Dynamics and Control*. Wiley.

Tanaskovic, M., Fagiano, L., and Gligorovski, V. (2019). Adaptive model predictive control for linear time varying MIMO systems. *Automatica*, 105, 237–245.

Wabersich, K.P. and Zeilinger, M.N. (2018). Safe exploration of nonlinear dynamical systems: A predictive safety filter for reinforcement learning. *arXiv:1812.05506*.