Preprints, 6th IFAC Conference on Nonlinear Model Predictive Control
Madison, WI, USA, August 19-22, 2018

Data-based predictive control via direct weight optimization ⋆

Jose R. Salvador * **D. Muñoz de la Peña** * **T. Alamo** *
A. Bemporad **

* *Dept. Ingeniería de Sistemas y Automática, Universidad de Sevilla, 41092 Sevilla, Spain (e-mail: {jrsalvador,dmunoz,talamo}@us.es).*
** *IMT School of Advanced Studies Lucca, 55100 Lucca, Italy (e-mail:alberto.bemporad@imtlucca.it).*

Abstract: In this paper we propose a novel data-based predictive control scheme in which the prediction model is obtained from a linear combination of past system trajectories. The proposed controller optimizes the weights of this linear combination taking into account simultaneously performance and the variance of the estimation error. For unconstrained systems, dynamic programming is used to obtain an explicit linear solution of a finite or infinite horizon optimal control problem. When constraints are taken into account, the controller needs to solve online a quadratic optimization problem to obtain the optimal weights, possibly considering also local information to improve the performance and estimation. A simulation example of the application of the proposed controller to a quadruple-tank system is provided.

Keywords: Predictive control, Database, Dynamic programming, Direct weight optimization, Model-free control.

1. INTRODUCTION

Model-free strategies have attracted attention from the control community in different paradigms. In model-free strategies, the only information available on the system is past trajectories which are directly used to achieve the final control objectives (Campi et al., 2002; Sala and Esparza, 2005; Hjalmarsson et al., 1998; van Heusden et al., 2010; Formentin et al., 2016; Tanaskovic et al., 2017).

In model predictive control (MPC), different data-driven strategies to obtain and adapt prediction models using ad-hoc identification and adaptive techniques have been proposed (Yang et al., 2015; Wang et al., 2007; Hou and Jin, 2011; Laurí et al., 2010) which need knowledge of the system to fit the structure of the process model. In (Kadali et al., 2003), a data-driven subspace approach is introduced to design the predictive controller. More recently reinforcement learning has been studied (Shah and Gopal, 2016),(Li et al., 2013). Another line of work is based on using machine learning techniques. In (Piga et al., 2018) data-driven direct controller synthesis are combined with MPC to control systems under input and output constraints without the need of a model of the open-loop process. Using similar techniques, model-free optimal control has been proposed in (Selvi et al., 2018). In (Aswani et al., 2013) some machine learning techniques are proposed to estimate the global uncertainty of the system, in order to improve predictions. In (Canale et al., 2012; Limon et al., 2017) nonlinear predictive controllers are designed based on data using Lipschitz interpolation techniques.

In this paper we propose to use a prediction model based on identification via direct weight optimization (Roll et al., 2005; Bravo et al., 2017) which are based on postulating an estimator that is linear in the observed outputs and then determining the weights in this estimator by direct optimization of a suitably chosen criterion. In particular, we propose to optimize the weights of this linear combination taking into account simultaneously performance and estimation error. For unconstrained systems, dynamic programming is used to obtain an explicit linear solution of a finite or infinite horizon optimal control problem. To obtain the explicit solution, a procedure to exploit the structure of the resulting quadratic optimization problem based on the matrix inversion lemma (Woodbury, 1950) is also provided. If constraints are taken into account, we propose to solve online a constrained quadratic optimization problem. In this case, the optimization problem can take into account local information to improve the performance.

2. PROBLEM FORMULATION

We consider a system described by an unknown time-invariant discrete-time model

$$z = f(x, u), \tag{1}$$

where $x \in X \subseteq \mathbb{R}^{n_x}$ is the system state, $u \in U \subseteq \mathbb{R}^{n_u}$ is the control input vector, $z \in X$ is the successor state, and f is the unknown transition function. In our setting, we will treat the system generating the data as if it were linear, although it may not be in reality if f is nonlinear.

The control objective is to regulate the system to the origin while minimizing the performance cost defined by

⋆ This work has been partially funded by the Spanish State Research Agency (AEI) and the European Regional Development Fund (ERFD) through the project DEOCS (ref. DPI2016-76493).

the following quadratic stage cost

$$L(x,u) = x^\top Q x + u^\top R u,$$

where Q and R are the tuning parameters of the controller.

If function f were known, this problem could be solved using off-the-shelf techniques, for example dynamic programming. Instead in this paper we propose to use the information stored in a database to predict the behavior of the system and solve simultaneously the control design problem. This database stores M different triplets of data corresponding to a state x_q, an input u_q, and the corresponding successor state z_q.

We assume that the samples collected in the data-set satisfy the following equation

$$z_q = f(x_q, u_q) + w_q, \tag{2}$$

where w_q models measurement errors that we assume independent from x_q, u_q.

Each data triplet (x_q, u_q, z_q) is denoted as a "candidate". The main idea is to estimate the dynamics of the system using a weighted linear combination of the candidates that is consistent with the current state and input (Roll et al., 2005; Bravo et al., 2017). The weight of each candidate is $\alpha_q \in \mathbb{R}$.

Lemma. Consider a set of M weights $\alpha_q \in \mathbb{R}$ and candidates (x_q, u_q, z_q) that satisfy (2) where (x_q, u_q) are independent for all q. If $f : X \times U \to X$ is linear and each entry w_{q_i} of vector w_q is a zero-mean, i.i.d. random variable with variance σ^2, then

$$f\left(\sum_{q=1}^{M} \alpha_q x_q, \sum_{q=1}^{M} \alpha_q u_q\right) = \sum_{q=1}^{M} \alpha_q z_q + e,$$

where the estimation error e is a random vector whose entries are zero-mean, i.i.d. random variables with variance $\sum_{q=1}^{M} \alpha_q^2 \sigma^2$.

Proof.

Since f is linear,

$$f\left(\sum_{q=1}^{M} \alpha_q x_q, \sum_{q=1}^{M} \alpha_q u_q\right) = \sum_{q=1}^{M} \alpha_q f(x_q, u_q).$$

From (2), if follows that

$$f(x_q, u_q) = z_q - w_q,$$

and hence

$$f\left(\sum_{q=1}^{M} \alpha_q x_q, \sum_{q=1}^{M} \alpha_q u_q\right) = \sum_{q=1}^{M} \alpha_q (z_q - w_q).$$

By defining $e = -\sum_{q=1}^{M} \alpha_q w_q$ we have that

$$f\left(\sum_{q=1}^{M} \alpha_q x_q, \sum_{q=1}^{M} \alpha_q u_q\right) = \sum_{q=1}^{M} \alpha_q z_q + e.$$

and taking into account that each entry w_{q_i} of vector w_q is a zero-mean, i.i.d. random variable with variance σ^2, then e is a random vector whose entries are zero-mean, i.i.d. random variables with variance $\sum_{q=1}^{M} \alpha_q^2 \sigma^2$. □

In the following section we propose to solve an optimization problem to determine the optimal candidates weights α_q to minimize both the performance cost and the estimation error variance following a predictive control approach based on dynamic programming.

3. UNCONSTRAINED EXPLICIT DATA-BASED PREDICTIVE CONTROL

In this section we present a data-based predictive control for unconstrained systems. The controller is based on solving a dynamic programming problem over a finite horizon in which the optimization variables are, at each iteration i, the optimal weights α_{iq} for each candidate q.

Following a dynamic programming approach for linear systems and quadratic costs, given a state x we assume that the cost to go is a quadratic function of the state,

$$J_i(x) = x^\top P_i x, \tag{3}$$

where P_0 defines the terminal cost of the data-based predictive controller. Then, $J_i(x)$ is defined recursively as follows:

$$J_{i+1}(x) = x^\top Q x + u^\top R u + J_i(z), \tag{4}$$

with

$$x = \sum_{q=1}^{M} \alpha_{iq}^*(x) x_q, \quad u = \sum_{q=1}^{M} \alpha_{iq}^*(x) u_q, \quad z = \sum_{q=1}^{M} \alpha_{iq}^*(x) z_q,$$

and

$$\begin{aligned} \alpha_{iq}^*(x) \; = \; & \arg \min_{\alpha_{i1} \ldots \alpha_{iM}} u^\top R u + J_i(z) + \beta \sum_{q=1}^{M} \alpha_{iq}^2 \\ & \text{s.t. } x = \sum_{q=1}^{M} \alpha_{iq} x_q, \\ & \quad u = \sum_{q=1}^{M} \alpha_{iq} u_q, \\ & \quad z = \sum_{q=1}^{M} \alpha_{iq} z_q, \end{aligned} \tag{5}$$

where $\beta > 0$ is a tuning parameter to provide a trade-off between performance and estimation error.

Note that the cost function of problem (5), which is used to calculate the optimal weights $\alpha_{iq}^*(x)$, includes a term that is proportional to the variance of the estimation error, $\sum_{q=1}^{M} \alpha_{iq}^2$. This term, however, is not included in the definition of the optimal cost of the proposed controller, $J_i(x)$.

Optimization problem (5) is solved recursively from $i = 1, \ldots, N$. The proposed unconstrained explicit data-based predictive control is then defined as

$$u^*(x) = \sum_{q=1}^{M} \alpha_{N-1,q}^*(x) u_q.$$

3.1 Multi-parametric solution of the proposed optimization problem

In this section, an explicit solution of problem (5) is provided using a multi-parametric approach in order to compute $J_{i+1}(x)$ from $J_i(x)$. To this end, we define the optimization vector

$$\alpha_i = [\alpha_{i1} \dots \alpha_{iM}]^\top \in \mathbb{R}^{M\times 1},$$

and the following matrices obtained from the database

$$\begin{aligned} X_q &= [\, x_1 \; x_2 \; \dots \; x_M \,], \\ U_q &= [\, u_1 \; u_2 \; \dots \; u_M \,], \\ Z_q &= [\, z_1 \; z_2 \; \dots \; z_M \,]. \end{aligned}$$

By treating x as a vector of parameters, optimization (5) is equivalent to the following parametric optimization problem

$$\begin{aligned} \alpha_i^*(x) = \arg\min_{\alpha_i} \; & \alpha_i^\top U_q^\top R U_q \alpha_i + \alpha_i^\top Z_q^\top P_i Z_q \alpha_i \\ & + \alpha_i^\top \beta_M \alpha_i \\ \text{s.t.} \quad & x = X_q \alpha_i, \end{aligned} \tag{6}$$

where $\beta_M = \beta I$.

Problem (6) can be rewritten as

$$\begin{aligned} \min_{\alpha_i} \; & \alpha_i^\top H_i \alpha_i \\ \text{s.t.} \; & T\alpha_i = Sx, \end{aligned} \tag{7}$$

where

$$\begin{aligned} H_i &= U_q^\top R U_q + Z_q^\top P_i Z_q + \beta_M \in \mathbb{R}^{M\times M}, \\ T &= X_q \in \mathbb{R}^{n_x \times M}, \\ S &= I \in \mathbb{R}^{n_x\times n_x}. \end{aligned}$$

If $H_i \succ 0$, the solution of problem (7) can be obtained solving by the following set of linear equations obtained from the Karush-Kuhn-Tucker conditions:

$$\begin{bmatrix} H_i & T^\top \\ T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \alpha_i \\ \lambda_i \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ Sx \end{bmatrix}, \tag{8}$$

where λ_i are the Lagrange multipliers of the equality constraints in (7).

Since $H_i \succ 0$, from (8) the optimal value of vector α_i can be obtained as a explicit function of λ_i,

$$\alpha_i^*(\lambda_i) = -H_i^{-1}T^\top \lambda_i. \tag{9}$$

From (8) and (9) we obtain that λ_i must satisfy the following equation

$$T(-H_i^{-1}T^\top\lambda_i) = Sx.$$

Assuming that the linear quadratic constraint qualification (LICQ) condition holds so that $TH_i^{-1}T^\top$ is invertible, we get

$$\lambda_i^*(x) = -(TH_i^{-1}T^\top)^{-1}Sx. \tag{10}$$

By substituting (10) in (9) we obtain

$$\alpha_i^*(x) = H_i^{-1}T^\top(TH_i^{-1}T^\top)^{-1}Sx = K_{i+1}x,$$

where

$$K_{i+1} = H_i^{-1}T^\top(TH_i^{-1}T^\top)^{-1}S. \tag{11}$$

Taking into account (3) and (4) it follows that

$$P_{i+1} = Q + K_i^\top(U_q^\top R U_q + Z_q^\top P_i Z_q)K_i. \tag{12}$$

Equation (12) provides an iterative procedure to obtain the value of P_{i+1} from P_i. The proposed unconstrained explicit data-based predictive control is then defined as

$$u^*(x) = U_q K_N x.$$

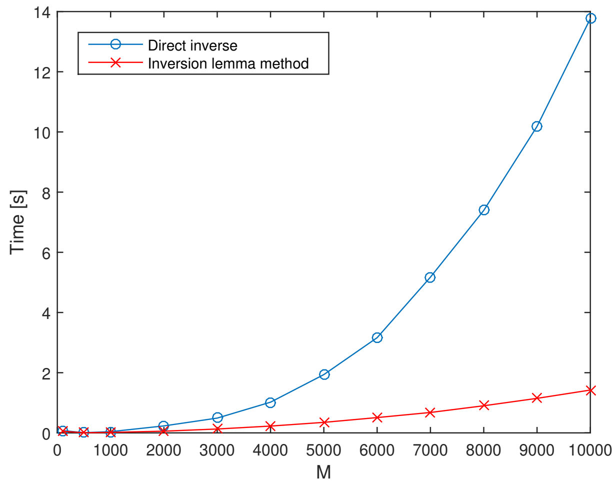

Fig. 1. Comparison between direct inverse of H_i and inverse obtained by the inversion lemma.

3.2 *Efficient computation of the inverse of H_i*

The computation of matrices K_{i+1} and P_{i+1} requires the computation of the inverse of H_i. This can be a cumbersome problem when the number of rows in the database M is high. We present next a procedure to compute this inverse efficiently profiting from the structure of the matrix.

Matrix H_i can be parameterized as follows

$$H_i = \beta_M - \Theta^\top \Upsilon \Theta,$$

where

$$\Theta = \begin{bmatrix} U_q \\ Z_q \end{bmatrix}, \qquad \Upsilon = -\begin{bmatrix} R & \mathbf{0} \\ \mathbf{0} & P_i \end{bmatrix}.$$

Considering the matrix inversion lemma, or the Sherman-Morrison-Woodbury formula, particularized for symmetric matrices, we have that

$$\begin{aligned} H_i^{-1} &= (\beta_M - \Theta^\top\Upsilon\Theta)^{-1} \\ &= \beta_M^{-1} + \beta_M^{-1}\Theta^\top(\Upsilon^{-1} - \Theta\beta_M^{-1}\Theta^\top)^{-1}\Theta\beta_M^{-1} \\ &= \frac{1}{\beta}I + \frac{1}{\beta^2}\Theta^\top(\Upsilon^{-1} - \frac{1}{\beta}\Theta\Theta^\top)^{-1}\Theta. \end{aligned}$$

The dimension of $(\Upsilon^{-1} - \frac{1}{\beta}\Theta\Theta^\top)$ is $(n_u + n_x)\times(n_u + n_x)$ which is much lower than the dimension M of H_i. This implies that using the inversion lemma is more efficient than a direct inversion. Figure 1 shows the comparison between the relation of computational time and M, matrix H_i dimension, for both techniques, direct inversion and inversion lemma method.

4. CONSTRAINED SINGLE-STEP DATA-BASED PREDICTIVE CONTROL

The proposed control-synthesis procedure can also be applied to constrained systems. However, in this case the resulting optimal control law would be a piece-wise linear function, and the corresponding optimal cost to go, a piece-wise quadratic function (Bemporad et al., 2002). This implies that a dynamic programming approach cannot be easily applied. Instead, a finite horizon open-loop constrained optimal control problem can be formulated and solved. In this case, for each prediction step j and candidate q a weight α_{jq} must be optimized so that for each prediction step $j \in \{1, \ldots, N-1\}$ the following constraints are satisfied in order to obtain a state and input prediction

$$x_j = \sum_{q=1}^{M} \alpha_{jq} x_q, \quad u_j = \sum_{q=1}^{M} \alpha_{jq} u_q, \quad z_j = \sum_{q=1}^{M} \alpha_{jq} z_q,$$

with $x_{j+1} = z_j$ and x_0 equal to the current measured state. Taking into account that the number M of possible candidates can be very large, we propose to use a prediction horizon of one step in order to consider state and input constraints, that we define as linear constraints by taking the sets X and U as polyhedra.

Solving an online optimization problem at each time step following a receding horizon approach allows us to take into account the current measured state in the definition of the optimization problem, in particular, in the estimation procedure. To this end we propose that, at each sampling time, only a subset of the candidates $G(x)$ is taken into account (for example, the closest candidates in the state space) and in addition, that the cost that depends on the square of the candidate weight for each candidate $\beta_q(x)$ depends also on the current state (for example, proportionally with the distance between the current state x and the candidate state x_q). Using close candidates and penalizing those that are far, in some sense, aims at obtaining a better local approximation of the system dynamics.

For a given state x, we propose to solve online the following optimization problem to determine the optimal candidates weights $\alpha_q^*(x)$ that minimize a cost function that depends on both the performance and the estimation error variance:

$$\begin{aligned} \min_{\alpha_q,\, q \in G(x)} \quad & x^\top Q x + u^\top R u + z^\top P z + \sum_{q=1}^{M} \beta_q(x) \alpha_q^2 \\ \text{s.t.} \quad & x = \sum_{q \in G(x)} \alpha_q x_q, \\ & u = \sum_{q \in G(x)} \alpha_q u_q \in U, \\ & z = \sum_{q \in G(x)} \alpha_q z_q \in X. \end{aligned} \tag{13}$$

The optimization problem (13) is solved at each sampling time k for the current state x. The proposed unconstrained explicit data-based predictive control law is then defined as

$$u^* = \sum_{q \in G(x)} \alpha_q^*(x) u_q.$$

5. EXAMPLE

In this section the proposed approach is applied to control a quadruple-tank system. The systems is made of four tanks that are filled from a storage tank located at the bottom of the plant through two three-way valves. The tanks at the top (tanks 3 and 4) discharge into the corresponding tank at the bottom (tanks 1 and 2, respectively). The inlet flows of the three-way valves are the manipulated variables of the real plant.

Table 1. Parameters of the plant.

	a_1	a_2	a_3	a_3	γ_a	γ_b
value	1.31e-4	1.51e-4	9.57e-4	8.82e-4	0.3	0.4
unit	m^2	m^2	m^2	m^2	-	-
	S	$h_{\max}$	$h_{\min}$	$q_{\max}$	$q_{\min}$	
value	0.06	1.3	0.3	3	0	
unit	m^2	m	m	m^3/h	m^3/h	

In order to generate data and test the controllers in closed-loop, a simulation model has been used. This model is based on the quadruple-tank process used in (Alvarado et al., 2011) and is given by the following differential equations:

$$\begin{aligned} S\dot{h}_1(t) &= -a_1\sqrt{2gh_1(t)} + a_3\sqrt{2gh_3(t)} + \gamma_a q_a(t), \\ S\dot{h}_2(t) &= -a_2\sqrt{2gh_2(t)} + a_3\sqrt{2gh_4} + \gamma_b q_b(t), \\ S\dot{h}_3(t) &= -a_3\sqrt{2gh_3(t)} + (1-\gamma_b) q_b(t), \\ S\dot{h}_4(t) &= -a_4\sqrt{2gh_4(t)} + (1-\gamma_a) q_a(t), \end{aligned} \tag{14}$$

where h_i and a_i with $i \in \{1, \ldots, 4\}$ refer to the water level and the discharge constant of tank i, respectively, S is the cross section of the tanks, q_j and γ_j with $j \in \{a, b\}$ denote the flow and the ratio of the three-way valve of pump j, respectively, and g is the gravitational acceleration. In this simulation we use the parameters of the quadruple tank process benchmark presented in (Alvarado et al., 2011) shown in Table 1 including maximum and minimum level and flow limits.

The level and flow variables define the following vectors

$$\mathbf{h} = [h_1 \quad h_2 \quad h_3 \quad h_4]^\top, \qquad \mathbf{q} = [q_a \quad q_b]^\top.$$

In order to design the proposed controller, a database of $M = 1000$ candidate triplets $(\mathbf{h}_q, \mathbf{q}_q, \mathbf{h}_{fq})$ is generated using the continuous-time model (14) and sampling every 5 seconds the solution,

$$\mathbf{h}_{fq} = \mathbf{h}(0) + \int_0^5 \frac{d\mathbf{h}(t)}{dt} dt + w_q$$

with $\mathbf{h}(0) = \mathbf{h}_q$, $\mathbf{q}(t) = \mathbf{q}_q$ for all t. Each entry w_{qi} of vector w_q is a zero-mean, i.i.d. random variable with uniform distribution inside the interval $[-0.01, 0.01]$ meters that models both measurement errors and perturbations. The level $\mathbf{h}_q$ and flow values $\mathbf{q}_q$ in each triplet are generated randomly between the maximum and minimum levels with a uniform distribution.

A tracking experiment is defined where a set of reference changes in the levels of the tanks, must be followed by manipulating the inlet flows. Each reference is defined by a target equilibrium point $(\mathbf{h}_r, \mathbf{q}_r)$ where

$$\mathbf{h}^r = [h_1^r \quad h_2^r \quad h_3^r \quad h_4^r]^\top, \qquad \mathbf{q}^r = [q_a^r \quad q_b^r]^\top,$$

with $r \in \{2, 3, 4\}$.

The initial levels in the experiment are 0.65 for all tanks. Reference changes occur every 3000 seconds. The initial reference is $(\mathbf{h}^2, \mathbf{q}^2)$, which changes to $(\mathbf{h}^4, \mathbf{q}^4)$, then to $(\mathbf{h}^3, \mathbf{q}^3)$ and then back to $(\mathbf{h}^2, \mathbf{q}^2)$. The references are the same used in the benchmark although in different order.

Table 2. Target references.

Reference r	2	3	4
h_1^r	0.30	0.50	0.90
h_2^r	0.30	0.75	0.75
h_3^r	0.30	0.30	1.06
h_4^r	0.30	1.18	0.58
q_a^r	1.11	2.20	1.53
q_b^r	1.35	1.36	2.54

The level and flow values of each reference are given in Table 2.

First, the proposed unconstrained explicit controller is applied. To this end, for each reference r a different controller is designed defining as state, input and prediction variables the deviation of the levels and flows from the corresponding target reference; that is,

$$x = \mathbf{h} - \mathbf{h}^r, \quad u = \mathbf{q} - \mathbf{q}^r, \quad z = \mathbf{h}_f - \mathbf{h}^r. \qquad (15)$$

By applying the change of variables of (15), for each reference r a different database including the M candidates (x_q^r, u_q^r, z_q^r) is calculated. These data are used to solve the finite horizon optimal control problem explicitly using a dynamic programming approach.

The controller design parameters are $Q = I$, $R = 0.01I$, $P = 10I$, $N = 4$ and $\beta_q = 1 \quad \forall\, q \in \{1, \ldots, M\}$. These parameters are used for the three different references. With a slight abuse of notation, we denote by P^r and K^r the matrices P_N and K_N for each reference r with $N = 4$. This implies that at each sampling time k, the controller is defined as

$$\mathbf{q}(k) = \mathbf{q}^r + U_q^r K^r(\mathbf{h}(k) - \mathbf{h}^r),$$

where r is the appropriate reference.

To carry out the simulation, the nonlinear model is used, including a bounded additive random perturbation in each level with uniform distribution inside the interval $[-0.01, 0.01]$. Figures 2 and 3 show the level and flow closed-loop trajectories for the unconstrained explicit controllers. It can be seen that the controller does not satisfy the input constraints, in particular, every time there is a reference change. In addition, from time 9000 to 9500 seconds, the level of the third tank falls below 0.3 meters during the transient.

Next, we apply the constrained predictive controller with prediction horizon one. For each reference, a different optimization problem is defined using the corresponding candidates (x_q^r, u_q^r, z_q^r) error variables and P^r as terminal cost in (13). In this simulation, the set of candidates taken into account at each sampling time k is obtained by selecting the 250 closest candidates to the current state x. In addition, for each candidate $q \in G(x))$, the parameter $\beta_q(x)$ is equal to the squared distance between the candidate and the current state, that is,

$$\beta_q(x) = (x(k) - x_q)^\top (x(k) - x_q)$$

Limiting the set of candidates and penalizing the weights of candidates far away from the current state may provide more smooth predictions (Roll et al., 2005; Bravo et al., 2017).

Figures 4 and 5 show the level and flow closed-loop trajectories for the constrained implicit controller. It can be seen that the level constraints are taken into account

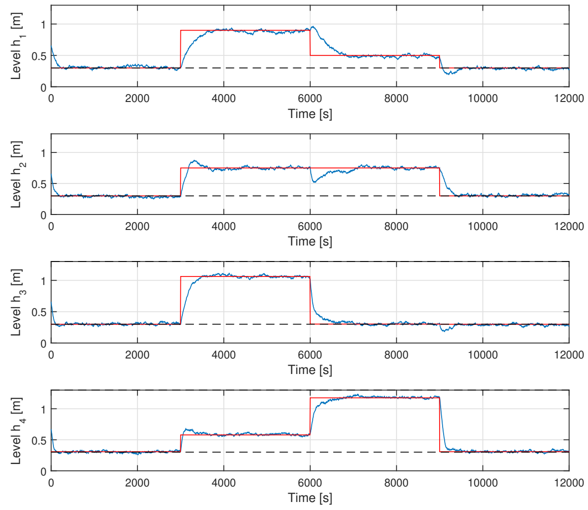

Fig. 2. Tank levels with unconstrained explicit controller.

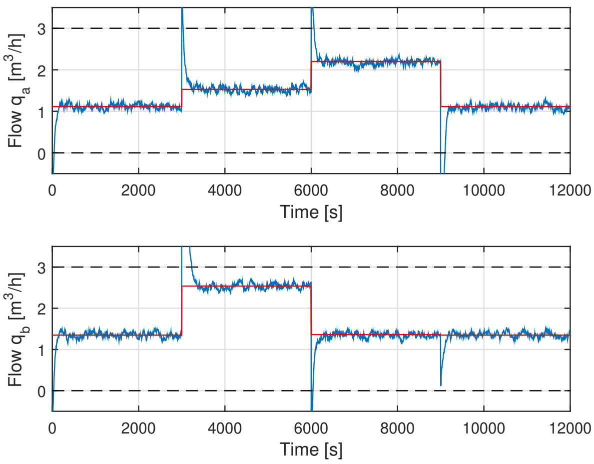

Fig. 3. Pump flows with unconstrained explicit controller.

and constraint violations are reduced when compared with the unconstrained controller.

The error of the direct weight optimization prediction for all the tank levels throughout this simulation is lower than 0.1 meters. This error depends on the deviation of the levels and flows from the corresponding target references considered. The mean value of the errors is lower than 0.001 meters. Finally, the same simulation has been carried out with a MPC based on the nonlinear model of the system to compare the performance. The difference between the predictive controller with horizon one based on direct weight optimization and the nonlinear controller is below 5%, although the number of sampling times in which the minimum level constraints are violated is lower in the simulation carried out by using the nonlinear MPC controller.

REFERENCES

Alvarado, I., Limon, D., Muñoz de la Peña, D., Maestre, J., Ridao, M., Scheu, H., Marquardt, W., Negenborn,

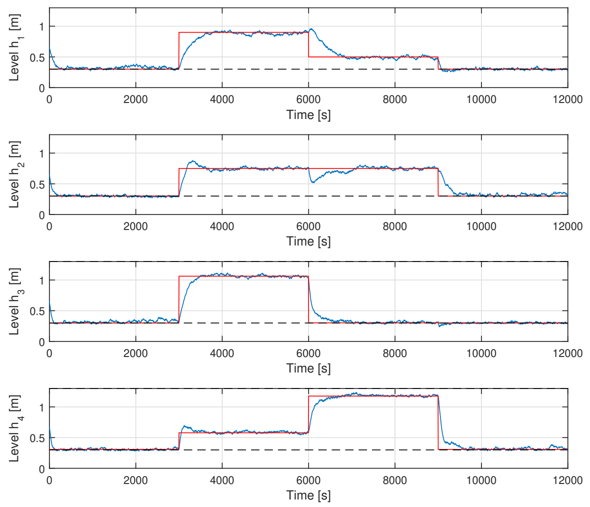

Fig. 4. Tank levels with constrained implicit controller.

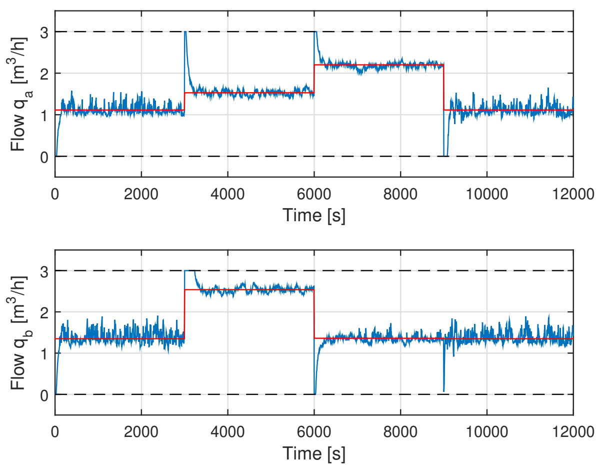

Fig. 5. Pump flows with constrained implicit controller.

R., De Schutter, B., Valencia, F., et al. (2011). A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark. *Journal of Process Control*, 21(5), 800–815.

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.

Bemporad, A., Morari, M., Dua, V., and Pistikopoulos, E.N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1), 3–20.

Bravo, J., Alamo, T., Vasallo, M., and Gegundez, M. (2017). A general framework for predictors based on bounding techniques and local approximation. *IEEE Transactions on Automatic Control*, 62(7), 3430–3435.

Campi, M., Lecchini, A., and Savaresi, S. (2002). Virtual reference feedback tuning: a direct method for the design of feedback controllers. *Automatica*, 38(8), 1337 – 1346.

Canale, M., Fagiano, L., and Signorile, M. (2012). Nonlinear model predictive control from data: a set membership approach. *International Journal of Robust and Nonlinear Control*, 24(1), 123–139.

Formentin, S., Piga, D., Tóth, R., and Savaresi, S.M. (2016). Direct learning of LPV controllers from data. *Automatica*, 65, 98 – 110.

Hjalmarsson, H., Gevers, M., Gunnarsson, S., and Lequin, O. (1998). Iterative feedback tuning: theory and applications. *IEEE Control Systems*, 18(4), 26–41.

Hou, Z. and Jin, S. (2011). Data-driven model-free adaptive control for a class of MIMO nonlinear discrete-time systems. *IEEE Transactions on Neural Networks*, 22(12), 2173–2188.

Kadali, R., Huang, B., and Rossiter, A. (2003). A data driven subspace approach to predictive controller design. *Control Engineering Practice*, 11(3), 261 – 278. Advances in Automotive Control.

Laurí, D., Rossiter, J., Sanchis, J., and Martínez, M. (2010). Data-driven latent-variable model-based predictive control for continuous processes. *Journal of Process Control*, 20(10), 1207 – 1219.

Li, D., Xi, Y., and Gao, F. (2013). Synthesis of dynamic output feedback RMPC with saturated inputs. *Automatica*, 49(4), 949 – 954.

Limon, D., Calliess, J., and Maciejowski, J. (2017). Learning-based nonlinear model predictive control. *IFAC-PapersOnLine*, 50(1), 7769 – 7776. 20th IFAC World Congress.

Piga, D., Formentin, S., and Bemporad, A. (2018). Direct data-driven control of constrained systems. *IEEE Transactions on Control Systems Technology*, 26(4), 1422–1429.

Roll, J., Nazin, A., and Ljung, L. (2005). Nonlinear system identification via direct weight optimization. *Automatica*, 41(3), 475–490.

Sala, A. and Esparza, A. (2005). Extensions to "virtual reference feedback tuning: A direct method for the design of feedback controllers". *Automatica*, 41(8), 1473 – 1476.

Selvi, D., Piga, D., and Bemporad, A. (2018). Towards direct data-driven model-free design of optimal controllers. In *Proceedings of the European Control Conference, Accepted for publication.*

Shah, H. and Gopal, M. (2016). Model-free predictive control of nonlinear processes based on reinforcement learning. *IFAC-PapersOnLine*, 49(1), 89 – 94. 4th IFAC Conference on Advances in Control and Optimization of Dynamical Systems ACODS 2016.

Tanaskovic, M., Fagiano, L., Novara, C., and Morari, M. (2017). Data-driven control of nonlinear systems: An on-line direct approach. *Automatica*, 75, 1 – 10.

van Heusden, K., Karimi, A., and Bonvin, D. (2010). Data-driven model reference control with asymptotically guaranteed stability. *International Journal of Adaptive Control and Signal Processing*, 25(4), 331–351.

Wang, X., Huang, B., and Chen, T. (2007). Data-driven predictive control for solid oxide fuel cells. *Journal of Process Control*, 17(2), 103 – 114.

Woodbury, M.A. (1950). Inverting modified matrices. *Memorandum Rept. 42*, 4, pp. Princeton University, Princeton, NJ.

Yang, J., Zheng, W.X., Li, S., Wu, B., and Cheng, M. (2015). Design of a prediction-accuracy-enhanced continuous-time MPC for disturbed systems via a disturbance observer. *IEEE Transactions on Industrial Electronics*, 62(9), 5807–5816.