# Direct learning of LPV controllers from data [⋆]

Simone Formentin [a], Dario Piga [b], Roland Tóth [c], Sergio M. Savaresi [a]

[a] *Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Piazza L. Da Vinci, 32, 20133 Milano, Italy.*
*E-mail: {simone.formentin,sergio.savaresi}@polimi.it*

[b] *IMT Institute for Advanced Studies Lucca, Piazza San Francesco 19, 55100 Lucca, Italy.*
*E-mail: dario.piga@imtlucca.it*

[c] *Department of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven,*
*The Netherlands. E-mail: r.toth@tue.nl*

## Abstract

In many control applications, it is attractive to describe nonlinear (NL) and time-varying (TV) plants by linear parameter-varying (LPV) models and design controllers based on such representations to regulate the behaviour of the system. The LPV system class offers the representation of NL and TV phenomena as a linear dynamic relationship between input and output signals, which relationship is dependent on some measurable signals, *e.g.*, operating conditions, often called as scheduling variables. For such models, powerful control synthesis tools are available, but the way how to systematically convert available first principles models to LPV descriptions of the plant, to efficiently identify LPV models for control from data and to understand how modeling errors affect the control performance are still subject of undergoing research. Therefore, it is attractive to synthesize the controller directly from data without the need of modeling the plant and addressing the underlying difficulties. Hence, in this paper, a novel data-driven synthesis scheme is proposed in a stochastic framework to provide a practically applicable solution for synthesizing LPV controllers directly from data. Both the cases of fixed order controller tuning and controller structure learning are discussed and two different design approaches are provided. The effectiveness of the proposed methods is also illustrated by means of an academic example and a real application based simulation case study.

*Key words:* Data-driven control, Identification for control, LPV systems, LS-SVM, Instrumental Variables.

## 1 Introduction

The concept of *linear parameter-varying* (LPV) systems, introduced in [36], offers a promising framework for modeling and control of a large class of *nonlinear* (NL) and *time-varying* (TV) systems. LPV systems can be seen as an extension of *linear time-invariant* (LTI) systems, with a linear dynamic relation between the input and the output signals. Unlike in the LTI case, these signal relations can change over the time and depend on a measurable time-varying signal, the so-called *scheduling variable*. Scheduling variables can be external signals like space coordinates or parameters used to describe changing operating conditions. In this way, the nonlinear and time-varying behavior of the system can be embedded in the solution set of a linear dynamic input-output relationship which varies with the scheduling variable [43]. The LPV modeling paradigm has evolved rapidly in the last two decades and has been applied in many applications like aircrafts [28], automotive systems [31,9,8], robotic manipulators [23] and induction motors [33].

Accurate and low complexity models of LPV systems can be efficiently derived from data using *input-output* (IO) representation based model structures [2,5,26,39,32], while state-space approaches appear to be affected by the curse of dimensionality and other approach-specific problems [48,47,49,12,45]. However, most of the control synthesis approaches are based on a state-space representation of the system dynamics (except a few recent works [1,7,50]) and state-space realization of complex IO models is theoretically solved, but difficult to accomplish in practice [39]. This transformation results in a state-minimal representation, which can have rational dependency on time-shifted versions of the scheduling signals. Alternative approaches can reduce the complexity of the scheduling-variable dependency, but at the price of a non state-minimal representation, for which

[⋆] This work was partially supported by the European Commission under project H2020-SPIRE-636834 “DISIRE - Distributed In-Situ Sensors Integrated into Raw Material and Energy Feedstock” and by the Netherlands Organization for Scientific Research (NWO, grant. no.: 639.021.127). A preliminary version of this work can be found in [19].

efficient model reduction is largely an open issue [41]. Moreover, the way the modeling error affects the control performance is unknown for most of the design methods and little work has been done on including information about the control objectives into the identification setting.

In this paper, a *method is proposed to design fixed-order LPV controllers in an IO form using directly the experimental data. In fact, this corresponds of designing controllers without deriving a model of the system.* This approach permits to avoid the critical (and time-consuming) approximation steps related to modeling, identification and state-space realization and it results in an automatic procedure in which only the desired closed-loop behavior has to be specified by the user. The proposed approach is developed for the case when the parametric structure of the controller is assumed to be given and also when the structure is needed to be selected (learnt) from data directly. The recent results in data-driven LPV model structure selection [42,27] employing *Least-Squares Support Vector Machines* (LS-SVM) [38] are exploited. In both cases, the controller synthesis problem is formulated as an optimization problem and *instrumental-variable* (IV) based identification techniques are used to efficiently cope with the noise affecting the signal measurements. The advantages of using an IV based approach are twofold: (i) it allows the design of the controller through convex optimization based on measured data from the system; (ii) the bias in the designed controller with respect to the optimal solution, due to the noise affecting the output measurements, is guaranteed to asymptotically converge to zero as the number of data samples increases.

Direct controller tuning approaches using a single set of IO data, also known as non-iterative data-driven control, have been already studied in the *linear time-invariant* (LTI) framework [3]. Well established approaches, like *Virtual Reference Feedback Tuning* (VRFT) [6,21] and *Non-iterative Correlation-based Tuning* (CbT) [44], have been widely discussed in the literature, see, *e.g.*, the recent results in [22,15,13,14,18,17]. Other recently introduced approaches are, *e.g.*, [34,16].

The first attempt to extend a data-driven method, namely the VRFT method, to LPV systems has been presented in [20], where data-driven gain-scheduled controller design has been proposed to realize a user-defined LTI closed-loop behavior. Although satisfactory performance has been shown for slowly varying scheduling trajectories, this methodology is far from being generally applicable to LPV systems. As a matter of fact, in the method presented in [20], the controller must be linearly parameterized and the reference behavior must be LTI. The latter requirement represents a strict limitation, since an LTI behavior might be difficult to realize in practice, as it may require too demanding input signals and dynamic dependence of the controller on the scheduling signal. On the other hand, the LPV extension of Non-iterative CbT has been found to be unfeasible, as the derivation of this approach is based on the commutation of the plant and the controller in the tuning scheme [24]. Unfortunately, such a commutation does not generally hold for parameter-varying transfer operators [43]. The recent work in [30] also deals with LPV direct data-driven control, but the framework is completely different from the one proposed herein. Specifically, in [30], the system is given in state-space form with measurable state vector, the optimal controller is assumed to be Lipschitz continuous and the whole method is developed in a deterministic set-membership setting. A direct data-driven LPV solution has been presented in a stochastic framework for feed-forward precompensator tuning in [4]. However, also in this case, no dynamic dependance is accounted for and the final objective is constrained to be LTI.

In summary, the main contributions of the paper are as follows: (i) a novel direct data-driven method is introduced for optimization of LPV controller parameters without the need of a model of the system to be controlled; the method is inspired by the VRFT concept, but it is entirely different from the straightforward LPV extension of VRFT in [20]; (ii) this is the first data-driven control method where learning the controller structure from data is achieved by the use of LS-SVM; (iii) we also show how inversion of a state-space reference model can be extended to the LPV case and how inversion up to a $k^{\text{th}}$-order delay can be achieved in case of no direct feedthrough. Finally, we compare the proposed approach in detail with other existing direct data-driven techniques. In this paper, we will focus on the SISO setting only. A non-straightforward MIMO extension of the results is possible by the use of so-called kernel representations, but it is beyond the scope of this work.

The paper is organized as follows. The formulation of the design problem is provided in Section 2, whereas Section 3 outlines the main idea behind the proposed methodology. Then, Section 4 illustrates the technical derivation of the method in case the structure of the controller is *a-priori* fixed (parametric design) and Section 5 deals with the case where also the controller parameterization has to be determined from data (nonparametric design). Section 6 discusses the proposed approach in comparison with the existing techniques. The effectiveness of the developed control design methodologies is shown by means of a numerical example in Section 7 and a real application based simulation study inspired by [25] in Section 8. The paper is ended by some concluding remarks.

## 2 Problem formulation

Let $\mathcal{G}_\mathrm{p}$ denote an unknown *single-input single-output* (SISO) LPV system described by the difference equation

$$A(p,t,q^{-1})y_\mathrm{o}(t) = B(p,t,q^{-1})u(t), \tag{1}$$

where $u(t) \in \mathbb{R}$ is the input signal, $y_\mathrm{o}(t) \in \mathbb{R}$ is the noise-free output and $p(t) \in \mathbb{P} \subseteq \mathbb{R}^{n_\mathrm{p}}$ is a set of $n_\mathrm{p}$ (exogenous) measurable scheduling variables. Without the loss of generality, in the sequel, the case of $n_\mathrm{p} = 1$ will be considered in order to keep the notation simple. In (1), $A(p,t,q^{-1})$ and $B(p,t,q^{-1})$ are polynomials in the backward time-shift operator $q^{-1}$ of finite degree $n_\mathrm{a}$ and $n_\mathrm{b}$, respectively, *i.e.*,

$$A(p,t,q^{-1}) = 1 + \sum_{i=1}^{n_\mathrm{a}} a_i(p,t)q^{-i}, \tag{2a}$$

$$B(p,t,q^{-1}) = \sum_{i=0}^{n_\mathrm{b}} b_i(p,t)q^{-i}, \tag{2b}$$

where the coefficients $a_i(p,t)$ and $b_i(p,t)$ are allowed to be nonlinear dynamic mappings of the scheduling sequence. In other words, such coefficients are not constrained to be (static) functions of $p(t)$, but they may also depend on $p(t-1)$, $p(t-2)$, $\ldots$, i.e., on finite many time-shifted values of $p(t)$. The measured output of the system is supposed to be corrupted by an additive, zero-mean, stationary colored noise $w(t)$, *i.e.*,

$$y(t) = y_\mathrm{o}(t) + w(t). \tag{3}$$

The system $\mathcal{G}_\mathrm{p}$ is assumed to be stable, where the notion of stability is defined as follows.

**Definition 1** *An LPV system, represented in terms of* (1)*, is called* stable *if, for all trajectories* $\{u(t), y(t), p(t)\}$ *satisfying* (1)*, with* $u(t) = 0$ *for* $t \geq 0$ *and* $p(t) \in \mathbb{P}$*, it holds that* $\exists\ \delta > 0$ s.t. $|y(t)| \leq \delta, \forall t \geq 0$.

**Remark** *Notice that, due to linearity, an LPV system that is stable according to Definition 1 also satisfies that*

$$\sup_{t\geq 0} |u(t)| < \infty \implies \sup_{t\geq 0} |y(t)| < \infty,$$

*for all* $p(t) \in \mathbb{P}$ *and* $\{u(t), y(t), p(t)\}$ *satisfying* (1)*. This property is known as Bounded-Input Bounded-Output (BIBO) stability in the* $\mathcal{L}_\infty$ *norm [39,11].*

The objective of the considered control problem is to achieve a desired closed-loop behavior $\mathcal{M}_\mathrm{p}$, given by a state-space representation

$$x_M(t+1) = A_M(p,t)x_M(t) + B_M(p,t)r(t), \tag{4a}$$

$$y_\mathrm{d}(t) = C_M(p,t)x_M(t) + D_M(p,t)r(t), \tag{4b}$$

where $y_\mathrm{d}$ denotes the desired closed-loop output for a given reference signal $r$. In the following, the operator $M(p,t,q^{-1})$ will be used as a shorthand form to indicate the mapping of $r$ to $y_\mathrm{d}$ in terms of (4), i.e., $y_\mathrm{d}(t) = M(p,t,q^{-1})r(t)$ for all trajectories of $p$ and $r$. In case the reference model is given in an IO form, this can be realized in a state-space representation using the so called maximally augmented realization form [40] or the approaches presented in [41].

Compared to (4), as we will see later, it is advantageous to express the parameterization of the controller to be synthesized in IO representation form. The class of controllers $\mathcal{K}_\mathrm{p}(\theta)$ is selected as

$$A_K(p,t,q^{-1},\theta)u(t) = B_K(p,t,q^{-1},\theta)(r(t) - y(t)), \tag{5}$$

where $A_K$ and $B_K$ are polynomials in $q^{-1}$ whose coefficients, parameterized with the design parameters collected in $\theta$, can be any (possibly dynamic) bounded function of the scheduling variable $p$.

**Remark** *The controller should be assumed to be dynamically dependent on* $p$ *in order to have enough flexibility to achieve the user-defined behavior. As a matter of fact, it is well-known that a static dependence would be a rather strong assumption for most real-world systems [39], [43].*

Assume that a collection of open-loop data $\mathcal{D}_N = \{u(t), y(t), p(t)\}$, $t \in \mathcal{I}_1^N = \{1, \ldots, N\}$, from (1) is available. Assume also that the following statements hold:

**A1.** there exists a value $\theta^\circ$ such that the controller $\mathcal{K}_\mathrm{p}(\theta^\circ)$ realizes $\mathcal{M}_\mathrm{p}$ in closed-loop;

**A2.** (5) is globally identifiable, *i.e.*, for two instances of the parameter vector $\theta$, namely $\theta^{(1)}$ and $\theta^{(2)}$, there exists a trajectory of $u$ and $p$ such that the response of the feedback interconnection of (1) and (5) is different if $\theta^{(1)} \neq \theta^{(2)}$. This implies that $\theta^\circ$ is unique;

**A3.** the dataset $\mathcal{D}_N$ is persistently exciting with respect to the used parameterization, *i.e.*, based on $\mathcal{D}_N$, $\theta^\circ$ can be uniquely determined.

Therefore, the model-reference control problem addressed in this paper can be formally stated as follows.

**Problem 1 (Model-reference control)** *Assume that a noisy dataset* $\mathcal{D}_N = \{u(t), y(t), p(t)\}_{t=1}^N$*, a reference model* (4) *and a controller class* $\mathcal{K}_\mathrm{p}(\theta)$ *as defined in* (5) *are given. Based on* $\mathcal{D}_N$*, determine the parameter vector* $\hat{\theta}$ *defining the controller* $\mathcal{K}_\mathrm{p}(\hat{\theta})$*, so that* $\hat{\theta}$ *asymptotically converges to* $\theta^\circ$*, as* $N \to \infty$.

**Remark** *Notice that, unlike in the LTI case, synthesis of a controller that achieves a user-defined behavior (i.e.,* *model-reference control) is not trivial in the LPV framework even from a model-based perspective. The main reason is that most of the techniques available for closed-loop model-matching cannot be extended to parameter-varying systems in the time-domain. Some results are instead available for $\mathcal{H}_2 - \mathcal{H}_\infty$ norm-based loop-shaping design in the frequency domain, e.g. [37], allowing model matching only with LTI reference models.*

## 3 Direct design from data

To start with, let the following additional assumption hold.

**A4.** $M(p,t,q^{-1})$ is invertible, where the inverse of a LPV mapping is defined as follows.

**Definition 2** *Given a causal LPV map $M(p,t,q^{-1})$ with input $r$, scheduling signal $p$ and output $y_\mathrm{d}$. The causal LPV mapping $M^\dagger(p,t,q^{-1})$ that gives $r$ as output when fed by $y_\mathrm{d}$, for any trajectory of $p$, is called the* left inverse *of $M(p)$, i.e., $M^\dagger(p,t,q^{-1})M(p,t,q^{-1}) = 1$.*

The computation of the *left inverse* of an LPV map is not straightforward; this problem will be discussed at the end of the section. Due to assumption **A1**, notice that Problem 1 can be reformulated as the optimization task (6) over a generic time interval $\mathcal{I}_1^N$, where $A(p,t)$ and $B(p,t)$ are identified from $\mathcal{D}_N$ and the argument $q^{-1}$ has been dropped for the sake of readability. Notice that this corresponds to matching the desired dynamical behavior with the actual closed-loop system, as graphically illustrated in Figure 1. If a consistent method is used for the identification of $A(p,t)$ and $B(p,t)$ (*e.g.*, the PEM method in [39]), and the polynomials are correctly parameterized, the estimate of the system asymptotically converges to the real $A(p,t)$ and $B(p,t)$ and the controller resulting as the solution of (6) makes the closed-loop system asymptotically converge to (4), thus solving Problem 1. However, such a model-reference problem is very hard to solve in the given form. In what follows, it will be shown that the problem can be further reformulated in a different fashion, which does not require neither to parameterize nor to identify the system $\mathcal{G}_\mathrm{p}$.

The proposed approach is based on two key ideas. The first one is that, *under Assumption* **A4**, *the dependence on the choice of $r$ can be annihilated.* As a matter of fact, by rewriting the first constraint of (6) as

$$r(t) = M^\dagger(p,t,q^{-1})\varepsilon(t) + M^\dagger(p,t,q^{-1})y_\mathrm{o}(t), \qquad (7)$$

where $M^\dagger$ denotes the left inverse of $M$, the optimization (6) can be reformulated as indicated in (8). Notice that, unlike in (6), the reference signal (7) is a projection of $y_\mathrm{o}$ and $\varepsilon$ to reconstruct $r$. Specifically, such a projection corresponds to the sum of two terms:

- the reference trajectory that would produce the data $y$ as an output, in case the closed-loop system is equal to $\mathcal{M}_\mathrm{p}$
- a term compensating the mismatch between $\mathcal{M}_\mathrm{p}$ and the actual closed-loop system, parameterized by $\theta$.

In this way, it even becomes indifferent whether $r$ exists or not in the real system, as (7) corresponds to a *virtual reference signal* satisfying the above given conditions. In other words, Problem (8) still corresponds to a closed-loop model matching task, but now it can be solved based on the open-loop data $\mathcal{D}_N$ (together with model information, since $\mathcal{G}_\mathrm{p}$ is still needed to compute the optimal solution).

Now we take another observation. *Since the available data in $\mathcal{D}_N$ is generated according to the system equations in the first and second constraints in* (8), *$\mathcal{D}_N$ can be used - under Assumption **A3** and $w = 0$ - as an alternative way to describe the dynamics of the system.*

Consider then the problem illustrated in (9) (underneath Equation (8)) where $u$, $y$ and $p$ are taken from the available dataset $\mathcal{D}_N$. Notice that such a problem is independent of the analytical description of $A(q^{-1},p)$ and $B(q^{-1},p)$ and therefore no model identification is needed to solve it. If the data are noiseless, *the global minimizer of* (9) *coincides with that of* (6), *providing the optimal controller achieving $\mathcal{M}_\mathrm{p}$ in closed-loop and yielding $\varepsilon = 0$.*

Let us now consider the practical situation. In case of noisy data (*i.e.*, $w \neq 0$), the estimate of the optimal controller is biased. As a matter of fact, the optimization procedure (9) pushes for $\varepsilon = 0$, whereas the minimizer of (8) would yield $\varepsilon$ such that

$$M^\dagger(p,t)\varepsilon(t) = M^\dagger(p,t)w(t) - w(t).$$

Proper stochastic treatment of the noise will be detailed in the next section based on the above presented idea for both the cases of known and unknown optimal controller structure. More specifically, the discrepancy between (8) and (9) will be dealt with, in order to obtain a solution of (9) that still asymptotically converges to the solution of (6).

Before proceeding, let us analyze the problem of computing $M^\dagger$. For reference maps given in the state-space form (4), the result of the following Proposition can be employed.

**Proposition 1** *Consider* (4) *and assume that there exists a $D_M^{-1}(p,t)$ which satisfies $D_M^{-1}(p,t)D_M(p,t) = 1$ and is bounded $\forall p \in \mathbb{P}^\mathbb{Z}$. Then,*

$$x_{M^\dagger}(t+1) = A_{M^\dagger}(p,t)x_{M^\dagger}(t) + B_{M^\dagger}(p,t)y_\mathrm{d}(t) \qquad (10a)$$

$$r(t) = C_{M^\dagger}(p,t)x_{M^\dagger}(t) + D_{M^\dagger}(p,t)y_\mathrm{d}(t) \qquad (10b)$$

$$
\begin{aligned}
&\min_{\theta,\varepsilon} \|\varepsilon\|_{\ell_2}^2 \\
&\text{s.t.} \qquad \varepsilon(t) = M(p,t)r(t) - y_{\mathrm{o}}(t), \qquad \forall t \in \mathcal{I}_1^N, \\
&A(p,t)y_{\mathrm{o}}(t) = B(p,t)u(t), \qquad \forall t \in \mathcal{I}_1^N, \\
&A_K(p,t,\theta)u(t) = B_K(p,t,\theta)(r(t) - y_{\mathrm{o}}(t)), \; \forall t \in \mathcal{I}_1^N.
\end{aligned} \tag{6}
$$

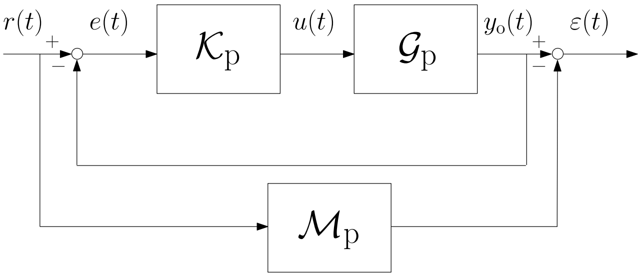
$r(t)$
$e(t)$
$\mathcal{K}_{\mathrm{p}}$
$u(t)$
$\mathcal{G}_{\mathrm{p}}$
$y_{\mathrm{o}}(t)$
$\varepsilon(t)$
+
−
+
−
$\mathcal{M}_{\mathrm{p}}$

Fig. 1. The proposed closed-loop behavior matching scheme. Note that the plant $\mathcal{G}_{\mathrm{p}}$, the controller $\mathcal{K}_{\mathrm{p}}$ and the reference model $\mathcal{M}_{\mathrm{p}}$ depend on the scheduling variable $p$.

*defines the left-inverse of* (4)*, where the matrix functions are given as*

$$
\begin{aligned}
A_{M^\dagger}(p,t) &= A_M(p,t) - B_M(p,t)D_M^{-1}(p,t)C_M(p,t), \\
B_{M^\dagger}(p,t) &= B_M(p,t)D_M^{-1}(p,t), \\
C_{M^\dagger}(p,t) &= -D_M^{-1}(p,t)C_M(p,t), \\
D_{M^\dagger}(p,t) &= D_M^{-1}(p,t).
\end{aligned}
$$

**Proof 1** *Consider the generic form of the output equation in* (10)*. Recalling* (4)*, the expression of* $y$ *becomes*

$$
\begin{aligned}
r(t) = C_{M^\dagger}(p,t)x_{M^\dagger}(t) + D_{M^\dagger}(p,t)C_M(p,t)x_M(t) + \\
+ D_{M^\dagger}(p,t)D_M(p,t)r(t).
\end{aligned} \tag{11}
$$

*Consider that* $x_{M^\dagger} = x_M$*. Since* $D_M$ *is assumed to be invertible, it follows that*

$$
D_{M^\dagger}(p,t) = D_M^{-1}(p,t),\; C_{M^\dagger}(p,t) = -D_M^{-1}(p,t)C_M(p,t).
$$

*Moreover, since* $x_{M^\dagger} = x_M$*, the state equations of* $M^\dagger$ *and* $M$ *can be combined such as*

$$
\begin{aligned}
A_{M^\dagger}(p,t)x_M(t) + B_{M^\dagger}(p,t)y_{\mathrm{d}}(t) = \\
A_M(p,t)x_M(t) + B_M(p,t)r(t),
\end{aligned} \tag{12}
$$

*where the left-hand term can be rewritten as*

$$
\begin{aligned}
A_{M^\dagger}(p,t)x_M(t) + B_{M^\dagger}(p,t)C_M(p,t)x_M(t) \\
+ B_{M^\dagger}(p,t)D_M(p,t)r(t).
\end{aligned} \tag{13}
$$

*Based on the above given equations, the expressions of* $A_{M^\dagger}$ *and* $B_{M^\dagger}$ *follow as*

$$
\begin{aligned}
A_{M^\dagger}(p,t) &= A_M(p,t) - B_{M^\dagger}(p)C_M(p,t), \\
B_{M^\dagger}(p,t) &= B_M(p,t)D_M^{-1}(p,t),
\end{aligned}
$$

*which completes the proof.* □

**Remark** *In case of* $D_M = 0$*,* $D_M$ *can be approximated by* $\hat{D}_M = \epsilon_D$ *where* $0 < \epsilon_D \ll 1$ *(as it is common in robust control [51]) to compute the inverse mapping. Another, more practical way to overcome the inverse in this case will be shown in Section 7.*

## 4 Data-driven parametric controller design

Suppose that the controller in (5) is such that

$$
A_K(p,t,q^{-1}) = 1 + \sum_{i=1}^{n_{\mathrm{a}_K}} a_i^K(p,t)q^{-i}, \tag{14a}
$$

$$
B_K(p,t,q^{-1}) = \sum_{i=0}^{n_{\mathrm{b}_K}} b_i^K(p,t)q^{-i}, \tag{14b}
$$

where

$$
a_i^K(p,t) = \sum_{j=1}^{n_i} a_{i,j}^K f_{i,j}(p,t), \tag{15a}
$$

$$
b_i^K(p,t) = \sum_{j=0}^{m_i} b_{i,j}^K g_{i,j}(p,t), \tag{15b}
$$

and $f_{i,j}(p,t)$ and $g_{i,j}(p,t)$ are *a-priori* chosen nonlinear (possibly dynamic) functions of the scheduling variable sequence $p$. The parameters $\theta$, characterizing the controller $\mathcal{K}_{\mathrm{p}}$, are herein the collection of the unknown constant terms $a_{i,j}^K$ and $b_{i,j}^K$, *i.e.*,

$$
\begin{aligned}
\theta = [\underline{a}_1^\top \; \dots \; \underline{a}_{n_{\mathrm{a}_K}}^\top \; \underline{b}_0^\top \; \dots \; \underline{b}_{n_{\mathrm{b}_K}}^\top]^\top, \\
\underline{a}_i = [a_{i,1}^K \; \dots \; a_{i,n_i}^K]^\top,\; \underline{b}_i = [b_{i,1}^K \; \dots \; b_{i,m_i}^K]^\top.
\end{aligned} \tag{16}
$$

**Remark** *Notice that, with such a parameterization, Problem* (9) *is generally non-convex because of the product between the optimization variables* $\varepsilon$ *and the parameters* $\theta$ *characterizing* $B_K(q^{-1},p,\theta)$*. Specifically, it is convex only if* $B_K(q^{-1},p,\theta)$ *is independent of* $\theta$*, whereas it is bi-convex in case of any linear dependance of* $B_K(q^{-1},p,\theta)$ *on* $\theta$*.*

$$\begin{aligned}&\min_{\theta,\varepsilon} \|\varepsilon\|_{\ell_2}^2\\ &\quad \text{s.t.} \qquad A(p,t)y_{\mathrm{o}}(t) \;=\; B(p,t)u(t), && \forall t\in\mathcal{I}_1^N,\\ &\qquad\quad A_K(p,t,\theta)u(t) \;=\; B_K(p,t,\theta)(M^\dagger(p,t)\varepsilon(t)+M^\dagger(p,t)y_{\mathrm{o}}(t)-y_{\mathrm{o}}(t)), && \forall t\in\mathcal{I}_1^N.\end{aligned} \tag{8}$$

$$\begin{aligned}&\min_{\theta,\varepsilon} \|\varepsilon\|_{\ell_2}^2\\ &\quad \text{s.t.} \quad A_K(p,t,\theta)u(t) \;=\; B_K(p,t,\theta)\big(M^\dagger(p,t)\varepsilon(t)+M^\dagger(p,t)y(t)-y(t)\big),\ \{u(t),y(t),p(t)\}\in\mathcal{D}_N.\end{aligned} \tag{9}$$

The controller parameters $\theta$ will be estimated on the basis of an *instrumental variable* scheme described in the following. The main reason for using the IV based approach is to replace the bi-convex optimization problem (9) with a stochastic considerations based convex problem, whose solution is guaranteed to asymptotically converge to the solution of (9) as the number of measurements goes to infinity.

Define the regressor $\phi(\xi_{\mathrm{o}},t)$ according to (19) (at the top of the next page), where the definition of the signal $\xi_{\mathrm{o}}(t)$ is

$$\xi_{\mathrm{o}}(t) = M^\dagger(p,t)y_{\mathrm{o}}(t) - y_{\mathrm{o}}(t), \tag{17}$$

and $\phi(\xi,t)$ similarly, with $\xi(t)$ as

$$\xi(t) = M^\dagger(p,t)y(t) - y(t). \tag{18}$$

Based on the above notation, the constraint in (9) can be rewritten as

$$u(t) - \phi^\top(\xi,t)\theta = B_K(p,t,\theta)M^\dagger(p,t)\varepsilon(t). \tag{20}$$

Consider now the optimization problem

$$\hat{\theta}_{\mathrm{IV}} = \arg\min_\theta \left\| \sum_{t=1}^N \zeta(t)\underbrace{\left(\phi^\top(\xi,t)\theta - u(t)\right)}_{B_K(p,t,\theta)M^\dagger(p,t)\varepsilon(t)} \right\|_2^2, \tag{21}$$

where $\zeta(t)$ is the *instrument*, a vector that has the dimension of $\phi(\xi,t)$ and is chosen by the user so that $\zeta(t)$ is not correlated with the noise-dependent term $\xi(t)-\xi_{\mathrm{o}}(t) = (M^\dagger(p,t)-1)w(t)$, *i.e.*, since $\mathbb{E}\{w(t)\}=0$,

$$\mathbb{E}\left\{\zeta(t)(M^\dagger(p,t)-1)w(t)\right\} = 0,\ \forall t\in\mathcal{I}_1^N. \tag{22}$$

Notice that (21) aims at the minimization of the projection (under the instrument $\zeta(t)$) of the colored residual $B_K(p,t,\theta)M^\dagger(p,t)\varepsilon(t)$ in (20). A possible way to construct an instrument with the above property in practice is to build another version of the regressor $\phi(\xi,t)$ by using a second experiment. Such an experiment has to be performed with the same input of the first experiment such that $\zeta(t)$ is correlated with the system dynamics, but it will contain a different realization of the noise, thus guaranteeing that (22) holds. The use of a solution based on *Refined Instrumental Variables* (RIV) [26] which does not require a second data set is currently under investigation.

By introducing the matrix notation

$$\Upsilon = [\zeta(1)\ \dots\ \zeta(N)]^\top,\ U = [u(1)\ \dots\ u(N)]^\top,$$
$$\Phi = [\phi(\xi,1)\ \dots\ \phi(\xi,N)]^\top,$$

Problem (21) can be also written in the compact form

$$\hat{\theta}_{\mathrm{IV}} = \arg\min_\theta \left\|\Upsilon^\top\left(\Phi\theta - U\right)\right\|_2^2, \tag{23}$$

whose solution is given by

$$\hat{\theta}_{\mathrm{IV}} = \left(\Upsilon^\top\Phi\right)^{-1}\Upsilon^\top U. \tag{24}$$

Notice that (24) only depends on the data and $M^\dagger$, whereas no information about the structure of $\mathcal{G}_{\mathrm{p}}$ or the noise model is required. The following result shows that the solution of (24) asymptotically converges to the solution of Problem (6), under the assumed noise conditions.

**Proposition 2** *If the optimal controller belongs to the model class specified in* (5), *then the controller parameters $\hat{\theta}_{\mathrm{IV}}$ in* (24) *asymptotically converge with probability 1 (w.p. 1) to the optimal parameters $\theta^\circ$ in* (6), *that is*

$$\lim_{N\to\infty}\hat{\theta}_{\mathrm{IV}} = \theta^\circ \quad w.p.\ 1. \tag{25}$$

**Proof 2** *Rewrite Equation* (24) *as*

$$\hat{\theta}_{\mathrm{IV}} = \left(\frac{1}{N}\Upsilon^\top\Phi\right)^{-1}\frac{1}{N}\Upsilon^\top U. \tag{26}$$

*The solution $\theta^\circ$ of Problem* (8) *- or equivalently the solution of Problem* (6) *- satisfies the following condition*

$$\begin{array}{llll} \phi(\xi_{\mathrm{o}},t) = [ & -u(t-1)f_{1,0}(p,t) & -u(t-1)f_{1,1}(p,t)\ldots & -u(t-1)f_{1,n_0}(p,t)\ldots \\ & -u(t-n_{\mathrm{a}_K})f_{n_{\mathrm{a}_K},0}(p,t) & -u(t-n_{\mathrm{a}_K})f_{n_{\mathrm{a}_K},1}(p,t)\ldots & -u(t-n_{\mathrm{a}_K})f_{n_{\mathrm{a}_K},n_{n_{\mathrm{a}_K}}}(p,t) \\ & \xi_{\mathrm{o}}(t)g_{0,0}(p,t) & \xi_{\mathrm{o}}(t)g_{0,1}(p,t)\ldots & \xi_{\mathrm{o}}(t)g_{0,m_1}(p,t)\ldots \\ & \xi_{\mathrm{o}}(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},0}(p,t) & \xi_{\mathrm{o}}(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},1}(p,t)\ldots & \xi_{\mathrm{o}}(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},m_{n_{\mathrm{b}_K}}}(p,t)\ ]^\top \end{array} \quad (19)$$

*(since, under the given assumptions, $\varepsilon = 0$ if and only if the optimum is achieved):*

$$U = \Phi_{\mathrm{o}}\theta^\circ, \quad (27)$$

*with*

$$\Phi_{\mathrm{o}} = [\phi(\xi_{\mathrm{o}},1) \ \ \ldots \ \ \phi(\xi_{\mathrm{o}},N)]^\top. \quad (28)$$

*Rewrite Equation (27) as*

$$U = \Phi\theta^\circ + \underbrace{\left(\tilde{\Phi} - \Phi\right)}_{\Delta\Phi}\theta^\circ. \quad (29)$$

*Note that the $t^{th}$ row $\Delta\phi_t$ of the matrix $\Delta\Phi$ is given by*

$$\begin{aligned} \Delta\phi_t = \big[0 \ \ldots \ 0 \ (M^\dagger(p)-1)w(t)g_{0,0}(p) \\ (M^\dagger(p)-1)w(t)g_{0,1}(p) \ \ldots \ (M^\dagger(p)-1)w(t)g_{0,m_1}(p) \\ \ldots \ (M^\dagger(p)-1)w(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},0}(p) \\ (M^\dagger(p)-1)w(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},1}(p) \ \ldots \\ \ldots \ (M^\dagger(p)-1)w(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},m_{n_{\mathrm{b}_K}}}(p)\ \big]. \quad (30) \end{aligned}$$

*Substitution of (29) into (26) leads to*

$$\begin{aligned} \hat{\theta}_{\mathrm{IV}} &= \left(\frac{1}{N}\Upsilon^\top\Phi\right)^{-1}\frac{1}{N}\Upsilon^\top\left(\Phi\theta^\circ + \Delta\Phi\theta^\circ\right) = \quad (31a) \\ &= \theta^\circ + \left(\frac{1}{N}\Upsilon^\top\Phi\right)^{-1}\frac{1}{N}\Upsilon^\top\Delta\Phi\theta^\circ. \quad (31b) \end{aligned}$$

*Since the instrument $\zeta(t)$ is not correlated with the entries of the matrix $\Delta\Phi$ (see Equations (22) and (30)),*

$$\lim_{N\to\infty}\frac{1}{N}\Upsilon^\top\Delta\Phi\theta^\circ = \mathbb{E}\{\Upsilon^\top\Delta\Phi\}\theta^\circ = 0 \quad (32)$$

*is satisfied w.p. 1. As a consequence, from (31) and (32), it follows that*

$$\lim_{N\to\infty}\hat{\theta}_{\mathrm{IV}} = \theta^\circ \quad \textit{w.p. } 1. \quad (33)$$

*This completes the proof.* □

## 5 Data-driven nonparametric controller design

To select the controller class such that the reference model is achievable is a hard task, when the model of the system is unknown. Therefore, in this part of the paper, the $p$-dependent coefficient functions $a_i^K(p,t)$ and $b_i^K(p,t)$ characterizing the LPV controller (14a)-(14b) are not *a-priori* parameterized, *i.e.*, the nonlinear basis functions $f_{i,j}(p,t)$ and $g_{i,j}(p,t)$ in (15) are not *a-priori* specified. In what follows, the problem will be analyzed in a primal form, whereas its solution will be provided in the, more suitable, dual form, according to the LS-SVM framework.

### 5.1 Primal problem

Let us write the $p$-dependent functions $a_i^K(p,t)$ and $b_i^K(p,t)$ in (14a) and (14b) as

$$a_i^K(p,t) = \theta_i^\top\psi_i(p,t) \quad (34a)$$

$$b_i^K(p,t) = \theta_{i+n_{\mathrm{a}_K}+1}^\top\psi_{i+n_{\mathrm{a}_K}+1}(p,t), \quad (34b)$$

where $i \in \mathcal{I}_1^{n_{\mathrm{a}_K}}$, $j \in \mathcal{I}_1^{n_{\mathrm{b}_K}}$ and $\theta_i \in \mathbb{R}^{n_{\mathrm{H}}}$ is a vector of unknown parameters and $\psi_i(p,t)$ (with $i \in \mathcal{I}_1^{n_{\mathrm{a}_K}+n_{\mathrm{b}_K}+1}$) is a nonlinear map from the original scheduling space $\mathbb{P}$ to an $n_{\mathrm{H}}$-dimensional space, commonly referred to as the *feature space*. Unlike the case of parametric controller design discussed in Section 4, neither the maps $\psi_i$ nor the dimension $n_{\mathrm{H}}$ of the vectors $\theta_i$ and $\psi_i$ are specified. Potentially, $\theta_i$ and $\psi_i(p,t)$ can be infinite-dimensional vectors (i.e., $n_{\mathrm{H}} = \infty$).

Let us define the vector $x \in \mathbb{R}^{n_{\mathrm{f}}}$ (with $n_{\mathrm{f}} = n_{\mathrm{a}_K} + n_{\mathrm{b}_K} + 1$) as follows:

$$x(\xi,t) = [-u(t-1) \ \ldots - u(t-n_{\mathrm{a}_K}) \ \ \xi(t) \ \ldots \ \xi(t-n_{\mathrm{b}_K})]^\top. \quad (35)$$

and let $x_i(\xi,t)$ be the $i$-th component of the vector $x(\xi,t)$.

Based on the introduced notation, the constraint in (9) can be rewritten in the regression form:

$$u(t) = \sum_{i=1}^{n_{\mathrm{f}}}\theta_i^\top\psi_i(p,t)x_i(\xi,t) + \underbrace{B_K(p,t,q^{-1})M^\dagger(p,t)\varepsilon(t)}_{\varepsilon_u(t)}. \quad (36)$$

Synthesis of the controller based on a data record $\mathcal{D}_N = \{u(t),y(t),p(t)\}_{t=1}^N$ is formulated in the LS-SVM framework [38] as the following convex optimization problem:

$$\min_{\theta_i,\varepsilon_u} \frac{1}{2}\sum_{i=1}^{n_\mathrm{f}} \theta_i^\top \theta_i \;+\; \frac{\gamma}{2N^2}\sum_{i=1}^{n_\mathrm{f}} \left\| \sum_{t=1}^{N} z_i(t)\varepsilon_u(t) \right\|_2^2 \tag{37a}$$

$$\text{s.t. } \varepsilon_u(t) = u(t) - \sum_{i=1}^{n_\mathrm{f}} \theta_i^\top \psi_i(p,t) x_i(\xi,t), \quad \forall t \in \mathcal{I}_1^N, \tag{37b}$$

where the instrument $z_i(t) \in \mathbb{R}^{n_\mathrm{H}}$ has the dimension of $\psi_i(p,t)$ (thus $z_i(t)$ can be an infinite-dimensional vector) and it has to be constructed to be uncorrelated with the noise term $\xi_\mathrm{o}(t) - \xi(t) = (M^\dagger(p,t) - 1)w(t)$, *i.e.*,

$$\mathbb{E}\{z_i(t)(M^\dagger(p,t)-1)w(t)\} = 0, \ \forall t \in \mathcal{I}_1^N, \ i = 1,\ldots,n_\mathrm{f}.$$

Note that, since the map $\psi_i(p,t)$ does not depend on the noise $w(t)$, the instrument $z_i(t)$ is constructed as follows:

$$z_i(t) = \psi_i(p,t) x_i(\hat{\xi},t), \tag{38}$$

where

$$x(\hat{\xi},t) = \Big[ -u(t-1) \ \ldots -u(t-n_{\mathrm{a}_K}) \ \ \hat{\xi}(t) \ \ldots \ \hat{\xi}(t-n_{\mathrm{b}_K}) \Big]^\top, \tag{39}$$

with $\hat{\xi}(t)$ being an approximation of the noise-free signal $\xi_\mathrm{o}(t)$ independent of the noise $w(t)$. This choice of the instrument is inspired by the LTI framework, where the instrument leading to the minimum variance estimate is given by the noise-free signal. For further details, see [26].

Note that two criteria are considered in problem (37). Specifically, the second term in the objective function of (37) aims at minimizing the (projection) of the residuals $\varepsilon_u(t)$, while the regularization term $\sum_{i=1}^{n_\mathrm{f}} \theta_i^\top \theta_i$ is included to prevent overfitting. In fact, since the dimension $n_\mathrm{H}$ of the parameter vector $\theta_i$ is not specified and it can be potentially infinite, penalizing the 2-norm of $\theta_i$ is essential to achieve an accurate estimate of the functions $a_i^K(p,t)$ and $b_i^K(p,t)$ in terms of the bias/variance trade-off. The regularization parameter $\gamma$ is tuned by the user, e.g., by using cross-validation to balance this trade-off. Let us introduce the matrix notation:

$$\Psi_i = \begin{bmatrix} \psi_i^\top(p,1) \\ \vdots \\ \psi_i^\top(p,N), \end{bmatrix}, \quad E = \begin{bmatrix} \varepsilon_u(1) \\ \vdots \\ \varepsilon_u(N) \end{bmatrix}, \tag{40a}$$

$$X_i(\xi) = \begin{bmatrix} x_i(\xi,1) & 0 & \cdots & 0 \\ 0 & x_i(\xi,2) & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & x_i(\xi,N) \end{bmatrix}. \tag{40b}$$

Then, Problem (37) can be written in the compact form:

$$\min_{\theta_i,\varepsilon_u} \frac{1}{2}\sum_{i=1}^{n_\mathrm{f}} \theta_i^\top \theta_i \;+\; \frac{\gamma}{2N^2}\sum_{i=1}^{n_\mathrm{f}} \left\| \Psi_i^\top X_i(\hat{\xi}) E \right\|_2^2 \tag{41a}$$

$$\text{s.t. } E = U - \sum_{i=1}^{n_\mathrm{f}} X_i(\xi)\Psi_i\theta_i. \tag{41b}$$

**Proposition 3** *The controller parameters $\hat{\theta}_{\mathrm{NP,IV}}$, obtained by minimizing Problem (41), asymptotically converge (w.p. 1) to:*

$$\lim_{N\to\infty} \hat{\theta}_{\mathrm{NP,IV}} = \theta^\circ - R^{-1}\gamma^{-1}\theta^\circ,$$

*where*

$$R = \lim_{N\to\infty} \gamma^{-1} I + \frac{1}{N^2}\Psi^\top Z Z^\top \Psi.$$

**Proof 3** *Note that Problem (37) can also be written as:*

$$\min_{\theta} \frac{1}{2}\theta^\top\theta + \frac{\gamma}{2N^2}\left\| Z^\top (U - \Psi\theta) \right\|_2^2, \tag{42a}$$

*with*

$$\theta = \begin{bmatrix} \theta_1^\top & \ldots & \theta_{n_\mathrm{f}}^\top \end{bmatrix}^\top, \tag{43a}$$

$$\Psi = \begin{bmatrix} \psi_1^\top(p,1)x_1(\tilde{\xi},1) & \ldots & \psi_{n_\mathrm{f}}^\top(p,1)x_{n_\mathrm{f}}(\tilde{\xi},1) \\ \ldots & \ldots & \ldots \\ \psi_1^\top(p,N)x_1(\tilde{\xi},N) & \ldots & \psi_{n_\mathrm{f}}^\top(p,N)x_{n_\mathrm{f}}(\tilde{\xi},N) \end{bmatrix}, \tag{43b}$$

$$Z = \begin{bmatrix} z_1^\top(1) & \ldots & z_{n_\mathrm{f}}^\top(1) \\ \ldots & \ldots & \ldots \\ z_1^\top(N) & \ldots & z_{n_\mathrm{f}}^\top(N) \end{bmatrix}. \tag{43c}$$

*The solution of Problem (42) is achieved for:*

$$\hat{\theta}_{\mathrm{NP,IV}} = \left(\gamma^{-1} I + \frac{1}{N^2}\Psi^\top Z Z^\top \Psi\right)^{-1} \frac{1}{N^2}\left(\Psi^\top Z Z^\top U\right). \tag{44}$$

*As already discussed in the proof of Proposition 2, the following condition holds:*

$$U = \Psi_\mathrm{o}\theta^\circ = \Psi\theta^\circ + \underbrace{(\Psi_\mathrm{o} - \Psi)}_{\Delta\Psi}\theta^\circ, \tag{45}$$

*where $\theta^\circ$ is the solution of Problem 6 and $\Psi_\mathrm{o}$ is defined as:*

$$\Psi_\mathrm{o} = \begin{bmatrix} \psi_1^\top(p,1)x_1(\xi_\mathrm{o},1) & \ldots & \psi_{n_\mathrm{f}}^\top(p,1)x_{n_\mathrm{f}}(\xi_\mathrm{o},1) \\ \ldots & \ldots & \ldots \\ \psi_1^\top(p,N)x_1(\xi_\mathrm{o},N) & \ldots & \psi_{n_\mathrm{f}}^\top(p,N)x_{n_\mathrm{f}}(\xi_\mathrm{o},N) \end{bmatrix}. \tag{46}$$

*Substitution of* (45) *in* (44) *leads to:*

$$\hat{\theta}_{\text{NP,IV}} = \theta^{\circ} +$$
$$\left(\gamma^{-1}I + \frac{1}{N^2}\Psi^{\top}ZZ^{\top}\Psi\right)^{-1}\left(\frac{1}{N^2}\Psi^{\top}ZZ^{\top}\Delta\Phi\theta^{\circ} - \gamma^{-1}\theta^{\circ}\right).$$

*As discussed in the proof of Proposition 2, the term* $\frac{1}{N}Z^{\top}\Delta\Phi$ *converges to* $0$ *w.p.* $1$ *as* $N$ *goes to infinity, thus*

$$\lim_{N\to\infty} \hat{\theta}_{\text{NP,IV}} = \theta^{\circ} - R^{-1}\gamma^{-1}\theta^{\circ}.$$

□

Note that the bias term $R^{-1}\gamma^{-1}\theta^{\circ}$ in the estimate of the controller parameters $\hat{\theta}_{\text{NP,IV}}$ does not depend on the realization of the noise $w(t)$, and it is only due to the regularization term $\frac{1}{2}\sum_{i=1}^{n_{\text{f}}}\theta_i^{\top}\theta_i$ used in Problem (41).

However, the controller parameters $\hat{\theta}_{\text{NP,IV}}$ minimizing (41) cannot be computed since an explicit representation of the feature maps $\psi_i(p,t)$ and of the instruments $z_i(t)$ would be needed. In order to compute both the parameters $\theta_i$ and the feature maps $\psi_i(p,t)$, the dual formulation of Problem (41) is considered next.

### 5.2 *Dual problem*

Let us define the *Lagrangian* $\mathcal{L}(\alpha,\theta,e)$ associated with Problem (41):

$$\mathcal{L}(\alpha,\theta,E) = \frac{1}{2}\sum_{i=1}^{n_{\text{f}}}\theta_i^{\top}\theta_i \; + \; \frac{\gamma}{2N^2}\sum_{i=1}^{n_{\text{f}}}\left\|\Psi_i^{\top}X_i(\hat{\xi})E\right\|_2^2 +$$
$$-\alpha^{\top}\left(E - U + \sum_{i=1}^{n_{\text{f}}}X_i(\xi)\Psi_i\theta_i\right),$$

where $\alpha \in \mathbb{R}^N$ is the collection of the *Lagrange multipliers*. Due to the affine nature of the constraints and convexity of the cost function, the global optimum of (41) is obtained when the *Karush-Kuhn-Tucker* (KKT) conditions reported in the following are fulfilled for all $i = 1,\ldots,n_{\text{f}}$:

$$\frac{\partial\mathcal{L}}{\partial\theta_i} = 0 \rightarrow \theta_i = \Psi_i^{\top}X_i(\xi)\alpha, \tag{47a}$$

$$\frac{\partial\mathcal{L}}{\partial E} = 0 \rightarrow \alpha = \frac{\gamma}{N^2}\sum_{i=1}^{n_{\text{f}}}X_i(\hat{\xi})\Psi_i\Psi_i^{\top}X_i(\hat{\xi})E, \tag{47b}$$

$$\frac{\partial\mathcal{L}}{\partial\alpha} = 0 \rightarrow E = U - \sum_{i=1}^{n_{\text{f}}}X_i(\xi)\Psi_i\theta_i. \tag{47c}$$

By substituting (47a) into (47c), we obtain:

$$E = U - \sum_{i=1}^{n_{\text{f}}}X_i(\xi)\Psi_i\Psi_i^{\top}X_i(\xi)\alpha. \tag{48}$$

Then, substitution of (48) into (47b) leads to:

$$\alpha = \frac{\gamma}{N^2}\sum_{i=1}^{n_{\text{f}}}X_i(\hat{\xi})\Psi_i\Psi_i^{\top}X_i(\hat{\xi})U +$$
$$-\frac{\gamma}{N^2}\sum_{i=1}^{n_{\text{f}}}X_i(\hat{\xi})\Psi_i\Psi_i^{\top}X_i(\hat{\xi})\sum_{j=1}^{n_{\text{f}}}X_j(\xi)\Psi_j\Psi_j^{\top}X_j(\xi)\alpha,$$

which has the solution

$$\alpha = R_{\text{D}}^{-1}(\Psi_i)\frac{1}{N^2}\sum_{i=1}^{n_{\text{f}}}X_i(\hat{\xi})\Psi_i\Psi_i^{\top}X_i(\hat{\xi})U, \tag{49}$$

with

$$R_{\text{D}}(\Psi_i) = \gamma^{-1}I + \frac{1}{N^2}\sum_{i=1}^{n_{\text{f}}}X_i(\hat{\xi})\Psi_i\Psi_i^{\top}X_i(\hat{\xi})\times$$
$$\times\sum_{j=1}^{n_{\text{f}}}X_j(\xi)\Psi_j\Psi_j^{\top}X_j(\xi). \tag{50}$$

The importance of the LS-SVM approach lies in the fact that the Lagrangian multipliers $\alpha$ can be computed without the proper knowledge of the feature maps $\psi_i(p,t)$ characterizing the matrix $\Psi_i$, as discussed in the following. Define the so-called *Grammian* matrix as $\Omega_i = \Psi_i\Psi_i^{\top}$. According to the Mercer's theorem [29,10], the Grammian matrices $\Omega_i$ (with $i = 1,\ldots,n_{\text{f}}$), can be defined in terms of kernel functions without the explicit knowledge of $\Psi_i$. More specifically, the generic $(t,k)$-th entry of $\Omega_i$, which is given by the inner product:

$$[\Omega_i]_{t,k} = \left\langle\psi_i(p,t),\psi_i(p,k)\right\rangle, \tag{51}$$

can be described by a positive definite *kernel* function $\kappa_i(p,t,k)$, i.e.,

$$[\Omega_i]_{t,k} = \left\langle\psi_i(p,t),\psi_i(p,k)\right\rangle = \kappa_i(p,t,k). \tag{52}$$

Specification of the kernels instead of the maps $\psi_i$ is called the *kernel trick* [46] and it provides the solution for (49) in terms of:

$$\alpha = R_{\text{D}}^{-1}(\Omega_i)\frac{1}{N^2}\sum_{i=1}^{n_{\text{f}}}X_i(\hat{\xi})\Omega_iX_i(\hat{\xi})U \tag{53}$$

with

$$R_{\text{D}}(\Omega_i) = \gamma^{-1}I + \frac{1}{N^2}\sum_{i=1}^{n_{\text{f}}}X_i(\hat{\xi})\Omega_iX_i(\hat{\xi})\sum_{j=1}^{n_{\text{f}}}X_j(\xi)\Omega_iX_j(\xi).$$

A typical choice of kernel, which provides uniformly effective representation of a large class of smooth functions, is the *Radial Basis Function* (RBF) kernel:

$$\kappa_i(p,t,k) = \exp\left(-\tfrac{\|p(t)-p(k)\|_2^2}{\sigma_i^2}\right), \tag{54}$$

where $\sigma_i > 0$ is a so-called hyper-parameter characterizing the width of the RBF and it is tuned by the user (e.g., through cross-validation). It is worth remarking that other positive definite kernels like linear, polynomial, rational, spline or wavelet kernels, can also be used to define the inner product between the feature maps $\psi_i$. Choosing the most appropriate kernel highly depends on the problem at hand. However, the analysis of the properties enjoyed by different types of kernels is out of the scope of the paper, therefore the reader is referred to [35] for further discussions on this issue.

Once the Lagrangian multipliers $\alpha$ are computed through (53), the $p$-dependent coefficient functions $a_i^K(p,t)$ and $b_i^K(p,t)$ characterizing the LPV controller (14a)-(14b) are obtained from (34) and (47a), *i.e.*,

$$\begin{aligned} a_i^K(\cdot) &= \psi_i^\top(\cdot)\theta_i = \\ &= \psi_i^\top(\cdot)\Psi_i^\top X_i(\xi)\alpha = \\ &= \sum_{t=1}^{N} \underbrace{\psi_i^\top(\cdot)\psi_i(p,t)}_{\kappa_i(p,t,\cdot)} x_i(\xi,t)\alpha_t, \end{aligned} \tag{55a}$$

$$\begin{aligned} b_i^K(\cdot) &= \psi_{i+n_{\mathrm{a}_K}+1}^\top(\cdot)\theta_{i+n_{\mathrm{a}_K}+1} = \\ &= \psi_{i+n_{\mathrm{a}_K}+1}^\top(\cdot)\Psi_{i+n_{\mathrm{a}_K}+1}^\top X_{i+n_{\mathrm{a}_K}+1}(\xi)\alpha = \\ &= \sum_{t=1}^{N} \underbrace{\psi_{i+n_{\mathrm{a}_K}+1}^\top(\cdot)\psi_{i+n_{\mathrm{a}_K}+1}(p,t)}_{\kappa_{i+n_{\mathrm{a}_K}+1}(p,t,\cdot)} x_{i+n_{\mathrm{a}_K}+1}(\xi,t)\alpha_t. \end{aligned} \tag{55b}$$

Note that the resulting controller coefficient functions only depend on the available observations in $\mathcal{D}_N$ and the specified kernel functions $\kappa_i(p,t,\cdot)$. On the other hand, the knowledge of the system dynamics and the feature maps $\psi_i(p,t)$ is not required.

## 6 Comparison with the existing techniques

As mentioned in the introduction, non-iterative data-driven methods already exist in the scientific literature and therefore a comparison with them is necessary to better clarify the novelty and the potential of the proposed approach. Specifically, the most well-known non-iterative methods: the VRFT [6] and the Non-iterative CbT [44] approaches are considered here.

The VRFT design scheme corresponds, in the noiseless LTI case, to the optimization problem

$$\hat{\theta}_{\mathrm{vr}} = \arg\min_{\theta,\varepsilon_{\mathrm{vr}}} \|\varepsilon_{\mathrm{vr}}\|_{\ell_2}^2, \tag{56}$$

where

$$\varepsilon_{\mathrm{vr}}(t) = u(t) - K(q^{-1},\theta)e_{\mathrm{v}}(t), \quad t \in \mathcal{I}_1^N,$$

$K(q^{-1},\theta)$ is the controller transfer function, $e_{\mathrm{v}}(t) = M^{-1}(q^{-1})y_{\mathrm{o}}(t) - y_{\mathrm{o}}(t)$ and $M(q^{-1})$ represents the transfer function of the LTI reference behavior. Such a signal can be seen as the error that would feed the controller in a "virtual" loop where the input $u$ of the identification experiment is the output of the controller and the complementary sensitivity function is $M(q^{-1})$. As a matter of fact, $e_{\mathrm{v}}$ is the difference between the "virtual" reference signal feeding the closed-loop system

$$r_{\mathrm{v}}(t) = M^{-1}(q^{-1})y_{\mathrm{o}}(t), \tag{57}$$

and the noiseless output of the experiment $y_{\mathrm{o}}(t)$. Although such a formulation might seem similar to the proposed approach in case of fixed $p$, there are a few significant differences:

- although in the strategy presented in this paper a "fictitious" reference signal, *i.e.*, (7), is computed to build a closed-loop optimization problem, such a reference is structurally different from the virtual reference signal (57) computed in VRFT (which is independent of the matching error);
- as indicated by the authors in [6], minimizing $\|\varepsilon_{\mathrm{vr}}\|_{\ell_2}^2$ is not exactly the same as minimizing $\|\varepsilon\|_{\ell_2}^2$, with $\varepsilon(t) = y_{\mathrm{o}}(t) - M(q^{-1})r(t)$, in the original model-matching problem;
- in standard VRFT, the denominator of the controller is *a-priori* fixed to guarantee a global solution, unlike in the proposed approach.

Moreover, in the LPV extension of VRFT [20],

- controllers are still linear in the parameters with a fixed denominator;
- no dynamic dependence on $p$ is taken into account, thus yielding a less general approach;
- the reference model needs to be LTI.

Regarding Non-iterative CbT, the differences with the proposed approach are even more evident. First of all, as indicated in [24], CbT cannot be extended to nonlinear systems, since the tuning scheme is based on the commutation of the plant and the controller (not allowed in the LPV setting). Moreover, the treatment of noise is based on extended instrumental variables minimizing a measure of the correlation between $u$ and $\varepsilon$.

However, it should be remarked that, in the LTI case, both VRFT and CbT consider also the case where **A1** does not hold. To handle this case, an asymptotical stability constraint and a bias-shaping prefilter are introduced. The analysis of this situation is obviously of great interest also in the LPV framework and therefore it is a critical objective of future research.

As far as the authors are aware, the study of this paper is the first in the field of non-iterative direct data-driven tuning where also the controller structure can be derived from data, whereas in VRFT and Non-iterative CbT the parameterization is *a-priori* fixed.

## 7 Numerical example

In this Section, the effectiveness of the proposed data-driven approach is demonstrated via a numerical example. The example is such that **A1** holds. In this way, not only the effectiveness of both the parametric and the nonparametric methods can be shown, but also a fair comparison between them can be carried out.

### *7.1 Plant description*

Consider the SISO LPV system $\mathcal{G}_\mathrm{p}$ defined as

$$\begin{aligned} x_G(t+1) &= p(t)x_G(t) + u(t) \\ y_\mathrm{o}(t) &= x_G(t) \\ y(t) &= y_\mathrm{o}(t) + w(t), \end{aligned} \tag{58}$$

where $p$ is an exogenous parameter taking values in $\mathbb{P} = [-0.4, 0.4]$. According to Definition 1, it can be easily shown that the system is stable for all possible trajectories of $p$.

### *7.2 Desired closed-loop behavior*

Let the desired behavior for the closed-loop system $\mathcal{M}_\mathrm{p}$ be given by the second order plant model

$$\begin{aligned} x_M(t+1) &= A_M(p,t)x_M(t) + B_M(p,t)r(t) \\ y_M(t) &= C_M(p,t)x_M(t) + D_M(p,t)r(t). \end{aligned} \tag{59}$$

where $y_M$ is the desired closed-loop trajectory for $y(t)$, $\Delta p(t) = p(t) - p(t-1)$ and

$$A_M(p,t) = \begin{bmatrix} -1 & 1 \\ -1-\Delta p(t) & 1 \end{bmatrix}, \; B_M(p,t) = \begin{bmatrix} 1+p(t) \\ 1+\Delta p(t) \end{bmatrix},$$
$$C_M = [1 \;\; 0]\,, \; D_M = 0,$$

This reference model is selected as it is achievable by using the simple controller parametrization defined in the next subsection.

### *7.3 Parametric controller design*

Assume now that an LPV PI controller $\mathcal{K}_\mathrm{p}$ of the form

$$\begin{aligned} x_K(t+1) &= x_K(t) + (\theta_0(p,t) + \theta_1(p,t))\,(r(t) - y(t)) \\ u(t) &= x_K(t) + \theta_0(p,t)\,(r(t) - y(t)) \end{aligned}$$

where

$$\theta_0(p,t) = \theta_{00} + \theta_{01}p(t), \tag{60}$$
$$\theta_1(p,t) = \theta_{10} + \theta_{11}p(t-1), \tag{61}$$

can be used for model reference control of $\mathcal{G}_\mathrm{p}$. The closed-loop dynamics can be written as a function of the controller parameters as:

$$x_F(t+1) = A_F(p,t)x_F(t) + B_F(p,t)r(t), \tag{62a}$$
$$y(t) = C_F(p,t)x_F(t) + D_F(p,t)r(t). \tag{62b}$$

where

$$A_F(p,t) = \begin{bmatrix} p(t) - \theta_0(p(t)) & 1 \\ -\theta_0(p(t)) - \theta_1(p(t)) & 1 \end{bmatrix},$$
$$B_F(p,t) = \begin{bmatrix} \theta_0(p(t)) \\ \theta_0(p(t)) + \theta_1(p(t)) \end{bmatrix},$$
$$C_F = [1 \;\; 0]\,, \; D_F = [0]\,.$$

By comparing $A_F$, $B_F$, $C_F$, $D_F$ and $A_M$, $B_M$, $C_M$, $D_M$, it appears that there exists a controller in the considered class which is able to achieve a closed-loop behavior equal to $\mathcal{M}_\mathrm{p}$, *i.e.*, **A1** holds. Specifically, the parameters of the optimal controller are such that

$$\theta_0^\circ(p,t) = \theta_{00}^\circ + \theta_{01}^\circ p(t) = 1 + p(t), \tag{63a}$$
$$\theta_1^\circ(p,t) = \theta_{10}^\circ + \theta_{11}^\circ p(t-1) = -p(t-1). \tag{63b}$$

This fact can be verified by substituting (63a) and (63b) into $A_F$ and $B_F$, which results in $A_M$ and $B_M$.

A parametric controller is computed through the approach proposed in Section 4, without deriving a model of $\mathcal{G}_\mathrm{p}$, so as to directly design from data the control law in its IO form (straightforwardly derivable from (60)):

$$\begin{aligned} u(t) =& u(t-1) + \theta_0(p,t)\,(r(t) - y(t)) + \\ &+ \theta_1(p,t)\,(r(t-1) - y(t-1))\,. \end{aligned} \tag{64}$$

For this purpose, a data set $\mathcal{D}_N$ of $N = 1000$ measurements are collected, by performing an experiment on (58), where $u(t)$ is selected as a white noise sequence with uniform distribution $\mathcal{U}(-1,1)$ and $p(t) = 0.4\sin(0.06\pi t)$. The output measurements are corrupted by a white noise sequence $w(t)$ with normal distribution $\mathcal{N}(0,\sigma^2)$ and standard deviation $\sigma = 0.2$. Under this experimental setting, the resulting *Signal to Noise Ratio* (SNR) is 9.8 dB. A second dataset with the same input has been generated for the construction of the instrument $\zeta$ as indicated in Section 4.

As a preliminary step, recall that $M^\dagger$ is needed to compute (18). Since $D_M$ is zero, the result of Proposition 1 cannot be used as it is. However, Proposition 1 can still become useful as follows. Suppose that there is no desired direct feedthrough ($D_M = 0$) between $r(t)$ and

$y_M(t)$, like in this example. However, similar to the concept of relative degree, it is possible to find a finite $k$ such that $y_M(t+k)$ directly depends on $u(t)$ for all $p(t) \in \mathbb{P}$. For $k > 0$, using straightforward calculations, Equation (65) holds. Now let us define $k$ to be the first value for which the left-hand-side (65) does depend on $r(t)$. Then, if follows that

$$y_M(t+k) = C_M(q^k p) \prod_{i=0}^{k-1} A_M(q^{k-i-1}p) x_M(t) + C_M(q^k p) \prod_{i=0}^{k-2} A_M(q^{k-i-1}p) B_M(p) r(t). \quad (66)$$

Hence, it is possible to redefine the reference system as $\mathcal{M}'_\mathrm{p}$ with

$$\begin{aligned} A_{M'}(p,t) &= A_M(p,t), \\ B_{M'}(p,t) &= B_M(p,t), \\ C_{M'}(p,t) &= C_M(q^k p) \textstyle\prod_{i=0}^{k-1} A_M(q^{k-i-1}p), \\ D_{M'}(p,t) &= C_M(q^k p) \textstyle\prod_{i=0}^{k-2} A_M(q^{k-i-1}p) B_M(p), \end{aligned}$$

According to the above definition for the system matrices, $y_{M'}(t) = y_M(t+k)$. If now $D_{M'} \neq 0$, $\forall p \in \mathbb{P}$, the inverse $\mathcal{M}'^{\dagger}_\mathrm{p}$ of $\mathcal{M}'_\mathrm{p}$ can be computed using Proposition 1 and, as a consequence, (18) is given by filtering $y$ with $\mathcal{M}'^{\dagger}_\mathrm{p}$ and shifting the data in time as

$$\tilde{\xi}(t) = M'^{\dagger}(p,t) y(t+k) - y(t), \ t \in \mathcal{I}_1^{N-1}. \quad (67)$$

It should be underlined here that, by doing so, the samples available for controller identification become $N-k$. This procedure is feasible because filtering is applied offline.

The controller parameters can now be computed using the direct data-driven method proposed in Section 4, *i.e.*, the IV estimation formula (24) using an instrument built with a second experiment (where the input sequence is the same as in the first test). The resulting values of the controller parameters are:

$$\theta_0(p,t) = 0.9852 + 1.0166 p(t), \quad (68a)$$
$$\theta_1(p,t) = -0.0153 - 0.9860 p(t-1), \quad (68b)$$

where small discrepancies with respect to (63a)-(63b) are due to the noise and the fact that $N$ is finite.

Despite these small variations, the controller appears to be effective in terms of matching of the desired closed-loop behavior. As an example, Fig. 2 illustrates a reference tracking (validation) experiment using a piecewise linear $p$, which is different from the trajectory of the estimation dataset $\mathcal{D}_N$. Note that in this closed-loop simulation, $y$ available for the synthesized controller is affected by measurement noise with the same characteristics as in the identification dataset.

### *7.4 Nonparametric controller design*

A nonparametric controller is now designed through the approach proposed in Section 5. The controller to be designed is given in the IO form:

$$\begin{aligned} u(t) = {} & a_1^K(p(t), p(t-1)) u(t-1) + \\ & + b_0^K(p(t), p(t-1)) \left(r(t) - y(t)\right) + \\ & + b_1^K(p(t), p(t-1)) \left(r(t-1) - y(t-1)\right), \end{aligned}$$

where the dependence of $a_1^K$, $b_0^K$ and $b_1^K$ on $p(t)$ and $p(t-1)$ is not *a-priori* specified (notice that neither the integral action is *a-priori* fixed). The functions $a_1^K$, $b_0^K$ and $b_1^K$ are estimated based on the same dataset used to design the parametric controller in Section 7.3. The values of the hyper-parameters $\gamma$ (eq. (37)) and $\sigma_i$ (characterizing the RBF $\kappa_i$ in (54)) are chosen through cross-validation. We remind that also the validation dataset is noisy. The obtained values of $\gamma$ and $\sigma_i$ are: $\gamma = 77844$ and $\sigma_i = 2.9$ for all $i = 1, 2, 3$. The realized closed-loop trajectory $y$ is plotted in Fig. 3. The obtained results show that, although no *a-priori* information on the dependence of the controller parameters $a_1^K$, $b_0^K$ and $b_1^K$ on the scheduling signal $p$ is used, the designed controller achieves similar performance to the parametric controller of Section 7.3. We should remark here that, no integral action has been directly enforced in the parametrization of the controller. Since, due to the data-driven nature of the approach, the matching of the closed-loop behavior is not perfect, which leads to a nonzero steady-state error of the response, see Fig. 3.

Obviously, the corresponding mean squared error computed for the dataset with $N_\mathrm{cl} = 400$ closed-loop samples, *i.e.*,

$$\mathrm{MSE} = \frac{1}{N_\mathrm{cl}} \sum_{t=1}^{N_\mathrm{cl}} \left(y(t) - y_M(t)\right)^2, \quad (69)$$

is a bit larger than in the parametric case, as shown in Table 1. This fact quantitatively illustrates the obvious correlation between the available preliminary knowledge and the achievable accuracy. It can be concluded that:

- the parametric approach gives the best performance provided that the structure of the optimal controller is *a-priori* available;
- the nonparametric approach should be preferred when the structure of the controller needs to be identified directly from data.

$$y_M(t+k) = C_M(q^k p)\prod_{i=0}^{k-1} A_M(q^{k-i-1}p)x_M(t) + C_M(q^k p)\sum_{j=0}^{k-2}\prod_{i=0}^{j} A_M(q^{k-i-1}p)B_M(q^{k-j-2}p)r(t+j) + C_M(q^k p)B_M(q^{k-1}p)r(t+k-1). \quad (65)$$

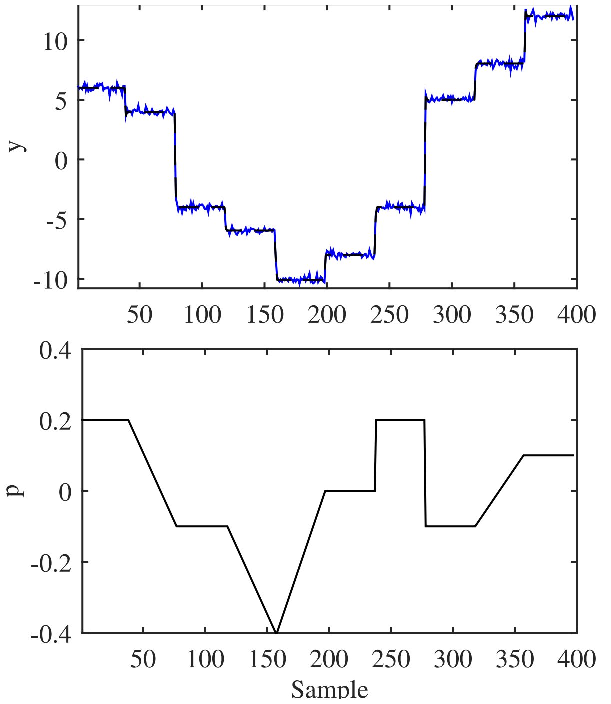

Fig. 2. Parametric controller design: (a) closed-loop output response $y$ (blue solid) and desired output response $y_M$ (black dashed), (b) scheduling signal $p$.

Table 1
Achieved MSE of the proposed approaches in the considered numerical example.

| | parametric | non-parametric |
|---|---|---|
| **MSE** | 0.0407 | 0.1565 |

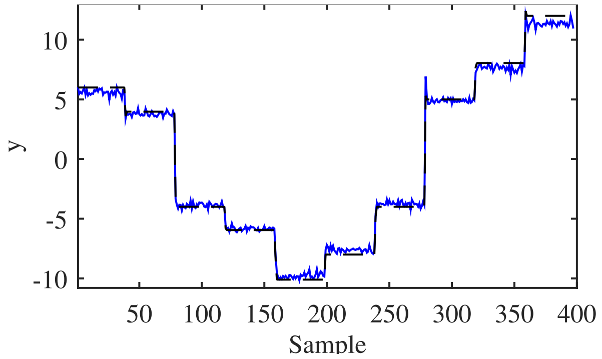

Fig. 3. Nonparametric controller design: closed-loop output response $y$ (blue solid) and desired output response $y_M$ (black dashed).

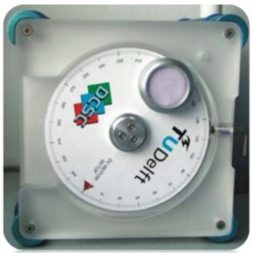

Fig. 4. The servo positioning system with unbalanced disk.

## 8 Servo-positioning system

As a more realistic case study, we investigate data-driven LPV control synthesis for the servo positioning system in Fig. 4, representing a voltage-controlled DC motor. The same example has already been used in other contributions, *e.g.*, [25]. The system has viscous friction and an additional mass is mounted on the disc connected to the rotor to make the mass distribution inhomogeneous. Note that in practical utilization of this application, an important control objective is to attenuate the load disturbance caused by the unbalanced disc and realize an LTI behavior of the closed-loop system, *i.e.*, hide the position dependent/nonlinear nature of the system from the user. Furthermore, unlike the previous numerical example, we will enforce a pure integral action in the controller parametrization in order to ensure zero steady state error.

### *8.1 System description*

The mathematical model of the considered servo-positioning system, used to simulate its dynamics behavior, is represented by the continuous-time state-space equation:

$$\begin{bmatrix}\dot{\theta}(\tau)\\ \dot{\omega}(\tau)\\ \dot{I}(\tau)\end{bmatrix} = \left(\begin{bmatrix}0 & 1 & 0\\ 0 & -\frac{b}{J} & \frac{K}{J}\\ 0 & -\frac{K}{L} & -\frac{R}{L}\end{bmatrix} + \begin{bmatrix}0 & 1 & 0\\ \frac{mgl}{J} & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\frac{\sin(\theta(\tau))}{\theta(\tau)}\right)\begin{bmatrix}\theta(\tau)\\ \omega(\tau)\\ I(\tau)\end{bmatrix} + \begin{bmatrix}0\\ 0\\ \frac{1}{L}\end{bmatrix}V(\tau), \quad (70)$$

with an output equation $y(\tau) = \theta(\tau)$. $V(\tau)$ [V] is the control input voltage over the armature, $I(\tau)$ [mA] is

Table 2
Physical parameters of the DC motor [25].

| | Description | Value |
|---|---|---|
| R | Motor resistance | 9.5 Ω |
| L | Motor inductance | $0.84{\cdot}10^{-3}$ H |
| K | Motor torque constant | $53.6{\cdot}10^{-3}$ Nm/A |
| J | Complete disk inertia | $2.2{\cdot}10^{-4}$ Nm$^2$ |
| b | Friction coefficient | $6.6{\cdot}10^{-5}$ Nms/rad |
| M | Additional mass | 0.07 kg |
| l | Mass distance from the center | 0.042 m |

the current, $\theta(\tau)$ [rad] is the angular position and $\omega(\tau)$ [rad/s] is the angular velocity of the disc. The parameters characterizing the DC motor are reported in Table 2. In terms of data acquisition, an ideal *zero-order-hold* (ZOH) setting is assumed where the actuation and output sampling is synchronized with a sampling period $T_\mathrm{s} = 0.01$ s and ideal anti-aliasing filter is assumed on the output measurements.

Note that, by choosing the output $\theta(\tau)$ as scheduling variable (i.e., $p(\tau) = \theta(\tau)$), the DC motor model (70) becomes an LPV model. Strictly speaking, it is a *quasi-LPV* model, as the scheduling variable is not an exogenous signal, but a state of the system. This indeed complicates the data-driven design control problem, as the scheduling signal in the dataset is also affected by the noise, and such noise is correlated with the noise on the output. However, we will show that in this application, where the SNR is typically high, the above fact does not affect the performance in a significant way.

### 8.2 Desired closed-loop behavior

The desired closed-loop behavior $\mathcal{M}_\mathrm{p}$ is described by the difference equations

$$\begin{aligned} x_M(t+1) &= A_M x_M(t) + B_M r(t), \\ \theta_M(t) &= C_M x_M(t) + D_M r(t). \end{aligned} \tag{71}$$

where $A_M = 0.99$, $B_M = 0.01$, $C_M = 1$ and $D_M = 0$. Namely, it is selected as a discrete-time linear time-invariant first-order plant, under the sampling time $T_s$ and a pole at (about) 0.15 Hz. From now on, we denote by $\theta_M(t)$ the desired closed-loop trajectory for the angular position of the disc $\theta(t)$.

### 8.3 Controller synthesis

We inject into the plant (70) a filtered white noise voltage signal with Gaussian distribution and standard deviation of 16 V and collect a data set $\mathcal{D}_N$ of $N = 1500$ input and output measurements (the input filter is selected as a first order digital filter with a cutoff frequency of 1.6 $Hz$). Notice that, unlike the previous example, the trajectory of the scheduling parameter cannot be arbitrarily selected here, as it corresponds to the achieved output signal. The output measurements are corrupted by a white noise sequence $w(t)$ with normal distribution and variance such that the SNR is 43 dB, which is realistic in terms of the considered encoder resolution. A second independent experiment with the same input is also performed, in order to build suitable instruments.

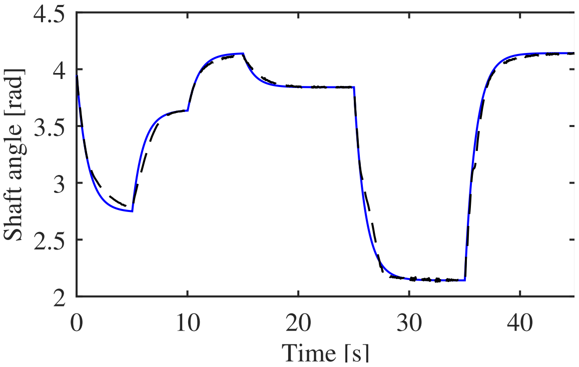

Fig. 5. Servo-positioning example: desired closed-loop response of the disc position $\theta_M$ (black dashed) and achieved closed-loop response with the synthesized controller (blue solid).

The controller is parametrized in the following form:

$$\begin{aligned} u(t) &= \sum_{i=1}^{4} a_i^K(\Pi(t))u(t-i) + \sum_{j=0}^{4} b_j^K(\Pi(t))e_\mathrm{int}(t-j) \\ e_\mathrm{int}(t) &= e_\mathrm{int}(t-1) + (r(t) - y(t)), \end{aligned}$$

where

$$\Pi(t) = [p(t-1) \;\; p(t-2) \;\; p(t-3) \;\; p(t-4)]^\top.$$

This means that we aim to design a fourth-order controller with integral action and nonparametric scheduling signal dependence. We remark that the use of the integrator is mandatory here as we need to achieve a closed-loop system with unitary gain.

By using the proposed non-parametric control synthesis approach with the gathered data, the values of the hyper-parameters $\gamma$ (see (37)) and $\sigma_i$ (see (54)) are found through cross-validation as $\gamma = 64163$ and $\sigma_i = 2.4$ for all $i = 1, 2, 3$. With the resulting controller, the response of the angular disc position $\theta$ to a multi-step excitation is compared to $\theta_M$ (computed for the same reference excitation) in Fig. 5.

Notice that, although in a realistic scenario an LTI closed-loop system is barely achievable (in fact, the performance is not the same for any step change in Fig. 5), the quality of the matching obtained with the proposed data-driven approach is satisfactory.

## 9 Concluding remarks

In this paper, a novel data-driven method has been introduced to directly design LPV model-reference controllers from IO data without the need of parameterizing, identifying and/or accomplishing state-space realization of an LPV model of the system. The method guarantees that the optimal controller achieving the reference closed-loop behavior is asymptotically obtained in case a feasible parameterization of the controller, which can realize the desired closed-loop behavior is *a-priori* known. It is worth remarking that commonly in practice, the model structure needed to include the optimal controller is not known. In such scenarios, the proposed non-parametric version of the approach can be employed, which allows to determine the controller structure directly from data. In both cases, the whole design procedure turns out to be equivalent to a single convex optimization problem.

The aim of this paper is to lay the basic foundations for future research in direct data-driven control of LPV plants, therefore much work still needs to be done, in order to make such an approach competitive with classical model-based design. Future activities will be especially devoted to the theoretical analysis of the cases where **A1** does not hold (the servo-positioning example presented in this paper has already shown inspiring results), variance analysis of the obtained control law and further studies of the proposed approach on real-world applications.

## References

[1] M. Ali, H. Abbas, and H. Werner. Controller Synthesis for Input-Output LPV Models. In *Proc. of the 49th Conference on Decision and Control, Atlanta, Georgia, USA*, pages 4018–4023, 2010.

[2] B. Bamieh and L. Giarre. Identification of linear parameter varying models. *International Journal of Robust and Nonlinear Control*, 12(9):841–853, 2002.

[3] A.S. Bazanella, L. Campestrini, and D. Eckhard. *Data-Driven Controller Design: The $\mathcal{H}_2$ Approach.* Springer, 2011.

[4] M. Butcher and A. Karimi. Data-driven tuning of linear parameter-varying precompensators. *International Journal of Adaptive Control and Signal Processing*, 24(7):592–609, 2009.

[5] M. Butcher, A. Karimi, and R. Longchamp. On the consistency of certain identification methods for linear parameter varying systems. In *Proc. of the 17th IFAC World Congress, Seoul, South Korea*, pages 4018–4023, 2008.

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

[7] V. Cerone, D. Piga, Regruto, and R. Tóth. Fixed order LPV controllers design for LPV models in input-output form. In *Proc. of the 51st Conference on Decision and Control, Maui, Hawaii, USA*, pages 6297–6302, 2012.

[8] V. Cerone, D. Piga, and D. Regruto. Set-membership LPV model identification of vehicle lateral dynamics. *Automatica*, 47(8):1794–1799, 2011.

[9] M. Corno, M. Tanelli, S. M. Savaresi, and L. Fabbri. Design and validation of a gain-scheduled controller for the electronic throttle body in ride-by-wire racing motorcycles. *IEEE Trans. on Control Systems Technology*, 19(1):18–30, 2011.

[10] F. Cucker and S. Smale. On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, 39(1):1–49, 2001.

[11] C. Desoer and M. Vidyasagar. *Feedback systems: input-output properties*, volume 55. SIAM, 1975.

[12] F. Felici, J. van Wingerden, and M. Verhaegen. Subspace identification of MIMO LPV systems using a periodic scheduling sequence. *Automatica*, 43(10):1684–1697, 2007.

[13] S. Formentin, A.L. Cologni, F. Previdi, and S.M. Savaresi. A data-driven approach to control of batch processes with an application to a gravimetric blender. *IEEE Transactions on Industrial Electronics*, 61(11):6383–6390, 2014.

[14] S. Formentin, M. Corno, S.M. Savaresi, and L. Del Re. Direct data-driven control of linear time-delay systems. *Asian Journal of Control*, 14(3):652–663, 2012.

[15] S. Formentin, P. De Filippi, M. Corno, M. Tanelli, and S.M. Savaresi. Data-driven design of braking control systems. *IEEE Transactions on Control Systems Technology*, 21(1):186–193, 2013.

[16] S. Formentin and A. Karimi. A data-driven approach to mixed-sensitivity control with application to an active suspension system. *IEEE Transactions on Industrial Informatics*, 9(4):2293–2300, 2013.

[17] S. Formentin and A. Karimi. Enhancing statistical performance of data-driven controller tuning via L2-regularization. *Automatica*, 50(5):1514–1520, 2014.

[18] S. Formentin, A. Karimi, and S.M. Savaresi. Optimal input design for direct data-driven tuning of model-reference controllers. *Automatica*, 49(6):1874–1882, 2013.

[19] S. Formentin, D. Piga, R. Tóth, and S.M. Savaresi. Direct data-driven control of linear parameter-varying systems. In *Proc. of the 52nd IEEE Conference on Decision and Control, Florence, Italy*, pages 4110 – 4115, 2013.

[20] S. Formentin and S. Savaresi. Virtual reference feedback tuning for linear parameter-varying systems. In *Proc. of the 18th IFAC World Congress, Milan, Italy*, pages 10219–10224, 2011.

[21] S. Formentin, S.M. Savaresi, and L. Del Re. Non-iterative direct data-driven controller tuning for multivariable systems: theory and application. *IET control theory & applications*, 6(9):1250–1257, 2012.

[22] S. Formentin, K. van Heusden, and A. Karimi. A comparison of model-based and data-driven controller tuning. *International Journal of Adaptive Control and Signal Processing*, 2013, DOI:10.1002/acs.2415.

[23] S. M. Hashemi, H. S. Abbas, and H. Werner. Low-complexity linear parameter-varying modeling and control of a robotic manipulator. *Control Engineering Practice*, 20(3):248–257, 2012.

[24] A. Karimi, K. Van Heusden, and D. Bonvin. Noniterative data-driven controller tuning using the correlation approach. In *European Control Conference*, pages 5189–5195, 2007.

[25] B. Kulcsar, J. Dong, J. van Wingerden, and M. Verhaegen. LPV subspace identification of a DC motor with unbalanced disc. In *IFAC Symposium on System Identification*, volume 15, pages 856–861, 2009.

[26] V. Laurain, M. Gilson, R. Tóth, and H. Garnier. Refined instrumental variable methods for identification of LPV Box-Jenkins models. *Automatica*, 46(6), 2010.

[27] V. Laurain, R. Tóth, W.X. Zheng, and M. Gilson. Nonparametric identification of LPV models under general noise conditions: an LS-SVM based approach. In *Proc. of the 16th IFAC Symposium on System Identification, Brussels, Belgium*, pages 1761–1766, 2012.

[28] B. Lu, F. Wu, and S. W. Kim. Switching LPV control of an F-16 aircraft via controller state reset. *IEEE Trans. on Control Systems Technology*, 14(2):267–277, 2006.

[29] J. Mercer. Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society of London, Series A*, 209:415–446, 1909.

[30] C. Novara. Direct design from data of LPV feedback controllers. In *Proc. of the 52nd IEEE Conference on Decision and Control, Florence, Italy*, pages 4098–4103, 2013.

[31] C. Novara, F. Ruiz, and M. Milanese. Direct identification of optimal SM-LPV filters and application to vehicle yaw rate estimation. *IEEE Trans. on Control Systems Technology*, 19(1):5–17, 2011.

[32] D. Piga, P. Cox, R. Tóth, and V. Laurain. LPV system identification under noise corrupted scheduling and output signal observations. *Automatica*, 53:329–338, 2015.

[33] E. Prempain, I. Postlethwaite, and A Benchaib. A Linear Parameter Variant $\mathcal{H}_\infty$ control design for an induction motor. *Control Engineering Practice*, 10(6):633–644, 2002.

[34] M.B. Radac, R.E. Precup, E.M. Petriu, S. Preitl, and C.A. Dragos. Data-driven reference trajectory tracking algorithm and experimental validation. *IEEE Transactions on Industrial Informatics*, 9(4):2327–2336, 2013.

[35] B. Schölkopf and A. Smola. *Learning with kernels.* Cambridge MA: MIT Press, 2002.

[36] J. S. Shamma and M. Athans. Analysis of gain scheduled control for nonlinear plants. *IEEE Trans. on Automatic Control*, 35(8):898–907, 1990.

[37] S. Skogestad and I. Postlethwaite. *Multivariable feedback control: analysis and design.* Wiley New York, 2007.

[38] J. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines.* World Scientific Publishing, Singapore, 2002.

[39] R. Tóth. *Modeling and identification of linear parameter-varying systems*, volume 403. Springer, 2010.

[40] R. Tóth. Maximum LPV-SS realization in a static form. *Technical report TUE-CS-2013-003, Department of Electrical Engineering, TU Eindhoven*, 2013.

[41] R. Tóth, H.S. Abbas, and H. Werner. On the state-space realization of LPV input-output models: Practical approaches. *IEEE Transactions on Control Systems Technology*, 20(1):139–153, 2012.

[42] R. Tóth, V. Laurain, W.X. Zheng, and K. Poolla. Model structure learning: A support vector machine approach for lpv linear-regression models. In *Proc. of the 50th IEEE Conference on Decision and Control and European Control Conference, Orlando, Florida, USA*, pages 3192–3197, 2011.

[43] R. Tóth, J.C. Willems, P.S.C. Heuberger, and P.M.J. Van den Hof. The behavioral approach to linear parameter-varying systems. *IEEE Transactions on Automatic Control*, 56(11):2499–2514, 2011.

[44] K. van Heusden, A. Karimi, and D. Bonvin. Data-driven model reference control with asymptotically guaranteed stability. *International Journal of Adaptive Control and Signal Processing*, 25(4):331–351, 2011.

[45] J. van Wingerden and M. Verhaegen. Subspace identification of bilinear and LPV systems for open-and closed-loop data. *Automatica*, 45(2):372–381, 2009.

[46] V. Vapnik. *Statistical Learning Theory.* Wiley-Interscience, 1998.

[47] V. Verdult, M. Lovera, and M. Verhaegen. Identification of linear parameter-varying state-space models with application to helicopter rotor dynamics. *International Journal of Control*, 77(13):1149–1159, 2004.

[48] V. Verdult and M. Verhaegen. Subspace identification of multivariable linear parameter-varying systems. *Automatica*, 38(5):805–814, 2002.

[49] V. Verdult and M. Verhaegen. Kernel methods for subspace identification of multivariable LPV and bilinear systems. *Automatica*, 41:1557–1565, 2005.

[50] S. Wollnack, H. Abbas, H. Werner, and R. Tóth. Fixed-structure LPV controller synthesis based on implicit input output representations. In *Proc. of the 52nd Conference on Decision and Control, Florence, Italy*, pages 2103–2108, 2013.

[51] K. Zhou and J. Doyle. *Essentials of robust control*, volume 180. Prentice hall Upper Saddle River, NJ, 1998.