Direct data-driven control of linear parameter-varying systems

Simone Formentin, Dario Piga, Roland Tóth and Sergio M. Savaresi

***Abstract*— In many control applications, nonlinear plants can be modeled as linear parameter-varying (LPV) systems, by which the dynamic behavior is assumed to be linear, but also dependent on some measurable signals, *e.g.*, operating conditions. When a measured data set is available, LPV model identification can provide low complexity linear models that can embed the underlying nonlinear dynamic behavior of the plant. For such models, powerful control synthesis tools are available, but the way the modeling error and the conservativeness of the embedding affect the control performance is still largely unknown. Therefore, it appears to be attractive to directly synthesize the controller from data without modeling the plant. In this paper, a novel data-driven synthesis scheme is proposed to lay the basic foundations of future research on this challenging problem. The effectiveness of the proposed approach is illustrated by a numerical example.**

I. INTRODUCTION

In many control applications, nonlinear plants can be modeled as *linear parameter-varying* (LPV) systems, where the dynamic behavior is characterized by linear relations which vary depending on some measurable time-varying signals, called scheduling signals. For example, the value of these variables can represent the actual operating point of the system. In the literature, it has been shown that accurate and low complexity models of LPV systems can be efficiently derived from data using *input-output* (IO) representation based model structures [10], while state-space approaches appear to be affected by the curse of dimensionality and other approach-specific problems [15]. However, most of the control synthesis approaches are based on a state-space representation of the system dynamics (except a few recent works [1] [5]) and state space realization of complex IO models is difficult to accomplish in practice. This transformation can result in a non minimal parameter dependency with time-shifted versions of the scheduling parameters or in a non state-minimal state-space realization, for which efficient model reduction is largely an open issue [12]. 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 *direct method is proposed to design fixed-order LPV controllers in an IO form using experimental data directly. 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 and state-space realization and it results in a fully automatic procedure, where only the desired closed-loop behavior has to be specified by the user. Moreover, although the optimization problem to solve the design of the controller is bi-convex in the general case, the final procedure turns out to be convex, when the problem is reformulated using suitable instrumental variables.

Simone Formentin is with the Department of Engineering, University of Bergamo, via Marconi 5, 24044 Dalmine, Italy. E-mail: simone.formentin@unibg.it

Dario Piga and Roland Tóth are with the Department of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513 5600 MB Eindhoven, The Netherlands. E-mail: {d.piga,r.toth}@tue.nl

Sergio M. Savaresi is with the Department of Electronics, Computer Science and Bioengineering, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano, Italy. E-mail: savaresi@elet.polimi.it

This work was partially supported by the Netherlands Organization for Scientific Research (NWO, grant. no.: 639.021.127).

Direct controller tuning using a single set of IO data, also known as non-iterative data-driven control, has been first studied in the *linear time-invariant* (LTI) framework [2]. Well established approaches have been introduced, like *Virtual Reference Feedback Tuning* (VRFT) [4] and *Non-iterative Correlation-based Tuning* (CbT) [14]. A first attempt to extend VRFT to LPV systems has been presented in [7], 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 [7], 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 [8]. Unfortunately, such a commutation does not generally hold for parameter-varying transfer operators [13]. A direct data-driven LPV solution has been presented for feed-forward precompensator tuning in [3]. Also in this case, no dynamic dependance is accounted for and the final objective is an LTI behavior.

In the remainder of this paper, a novel data-driven scheme for LPV controller synthesis without the need of a model of the system is presented. The formulation of the design problem is provided in Section II, whereas Section III and Section IV illustrate the technical development of the method for noiseless and noisy data, respectively. Section V compares the proposed scheme to existing techniques, whereas the effectiveness of the introduced method is demonstrated

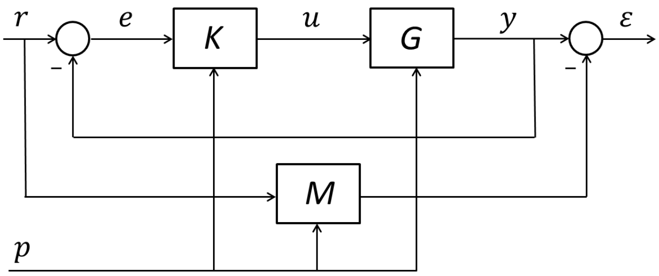

Fig. 1. Data-driven LPV control configuration: the proposed closed-loop behavior matching scheme.

by a simulation example in Section VI. Some final remarks end the paper.

II. PROBLEM FORMULATION

Consider the one *degree-of-freedom* (DOF) control architecture depicted in Figure 1. Let G denote an unknown *single-input single-output* (SISO) LPV system described by the difference equation

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

where $u(t) \in \mathbb{R}$ is the input signal, $y(t) \in \mathbb{R}$ is the noise-free output and $p(t) \in \mathbb{P} \subseteq \mathbb{R}^{n_{\mathrm{p}}}$ is a set of n_{p} (exogenous) measurable scheduling variables. From now on, for simplicity, the case of $n_{\mathrm{p}} = 1$ will be considered.

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_{a} and n_{b}, respectively, *i.e.*,

$$A(p,t,q^{-1}) = 1 + \sum_{i=1}^{n_{\mathrm{a}}} a_i(p,t)q^{-i},$$
$$B(p,t,q^{-1}) = \sum_{i=0}^{n_{\mathrm{b}}} b_i(p,t)q^{-i},$$

where the coefficients $a_i(p,t)$ and $b_i(p,t)$ are nonlinear (possibly dynamic) mappings of the whole scheduling sequence, *i.e.*, $p(t)$, $p(t-1)$ and so on.

The system G 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 $u(t) = 0,\ t \geq 0$, it holds that $\exists\ \delta > 0$ s.t. $|y(t)| \leq \delta, \forall t \geq 0$. ■

Remark 1: 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 $\{u(t), y(t), p(t)\}$ satisfying (1). This property is known as *Bounded-Input Bounded-Output* (BIBO) *stability* in the $\mathcal{L}_\infty$ norm [10]. ■

Consider that, as the objective of the control design, a desired closed-loop behavior is given by a state-space representation

$$\begin{aligned} x_M(t+1) &= A_M(p,t)x_M(t) + B_M(p,t)r(t), \\ y(t) &= C_M(p,t)x_M(t) + D_M(p,t)r(t). \end{aligned} \tag{2}$$

In the following, the transfer operator $M(p,t,q^{-1})$, which indicates the infinite impulse response of the reference model (2) will be used as a shorthand form to indicate the mapping of r via M. Formally, M is such that $y(t) = M(p,t,q^{-1})r(t)$ for all trajectories $\{u(t), y(t), p(t)\}$ satisfying (2). 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 [11] or the approaches presented in [12].

Furthermore, consider that the controller K, parameterized through θ, can be represented as

$$A_K(p,t,q^{-1},\theta)u(t) = B_K(p,t,q^{-1},\theta)(r(t) - y(t)), \tag{3}$$

where

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

and

$$a_i^K(p,t) = \sum_{j=1}^{n_0} a_{i,j}^K f_{i,j}(p,t),\ b_i^K(p,t) = \sum_{j=0}^{m_i} b_{i,j}^K g_{i,j}(p,t),$$

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

$$\begin{aligned} \theta &= [\underline{a}_1^\top\ \ldots\ \underline{a}_{n_{\mathrm{a}_K}}^\top\ \underline{b}_0^\top\ \ldots\ \underline{b}_{n_{\mathrm{b}_K}}^\top]^\top, \\ \underline{a}_i &= [a_{i,1}^K\ \ldots\ a_{i,n_i}^K]^\top,\ \underline{b}_i = [b_{i,1}^K\ \ldots\ b_{i,m_i}^K]^\top. \end{aligned} \tag{4}$$

Remark 2: The controller has been 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 of real-world systems [10], [13]. ■

Assume now that a collection of open-loop data $\mathcal{D}_N = \{u(t), y_w(t), p(t)\},\ t \in \mathcal{I}_1^N = \{1, \ldots, N\}$, is available, wherein

$$y_w(t) = y(t) + w(t) \tag{5}$$

and $w(t)$ represents a zero-mean colored output noise. Specifically,

$$D(p,t,q^{-1})w(t) = C(p,t,q^{-1})v(t),$$

where v is a zero mean white noise of unit variance, $D(p,t,q^{-1})$ and $C(p,t,q^{-1})$ are polynomials in q^{-1} of finite degree n_{d} and n_{c}, respectively, *i.e.*,

$$C(p,t,q^{-1}) = 1 + \sum_{i=1}^{n_{\mathrm{c}}} c_i(p,t)q^{-i},$$
$$D(p,t,q^{-1}) = 1 + \sum_{i=1}^{n_{\mathrm{d}}} d_i(p,t)q^{-i},$$

and the coefficients $c_i(p,t)$ and $d_i(p,t)$ are unknown nonlinear (possibly dynamic) mappings of the the scheduling parameter sequence p. The model-reference control problems considered in this paper can then be stated as follows.

Problem 1 (Design with noiseless data): Assume that a noiseless dataset $\mathcal{D}_N = \{u(t), y(t), p(t)\}$, $t \in \mathcal{I}_1^N$, a reference model (2) and a controller structure (3) are given. Determine θ, so that the closed-loop system composed by (1) and (3) is equal to (2). ■

Problem 2 (Design with noisy data): Assume that a noisy dataset $\mathcal{D}_N = \{u(t), y_w(t), p(t)\}$, $t \in \mathcal{I}_1^N$, a reference model (2) and a controller structure (3) are given. Determine θ, so that the closed-loop system composed by (1) and (3) asymptotically converges to (2), as $N \to \infty$. ■

First, for the clarity of the exposition, Section III will be dedicated to the unrealistic (but simpler) Problem 1, for which the key ideas of the approach will be introduced. Then, the solution of the realistic (but more complex) Problem 2 will be developed as an extension of the noiseless case in Section IV.

Remark 3: Notice that, unlike in the LTI case, designing 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. ■

III. LPV CONTROLLER TUNING FROM DATA: NOISELESS DATA

In this Section, Problem 1 will be addressed. Notice that the objective can be interpreted as an optimization problem over a generic time interval $\mathcal{I}_1^N$, described by (6).

As a first step, assume that the following statements hold:

A1. the objective can be achieved, *i.e.*, there exists a value of θ such that the closed-loop behavior is equal to $M(p,t)$ for any trajectory of p;

A2. $M(p,t)$ is invertible;

where the inverse of a LPV mapping Σ is defined as follows.

Definition 2: Given a causal LPV map Σ with input x_1, scheduling signal p and output x_2. The causal LPV mapping $\Sigma^\dagger$ that gives x_1 as output when fed by x_2, for any trajectory of p, is called the *left inverse* of Σ. ■

The proposed approach is based on two key ideas. The first one is that, *under assumption **A2**, the dependence on the choice of r can be removed*. As a matter of fact, by rewriting the first constraint of (6) as

$$r(t) = M^\dagger(p,t,q^{-1})\varepsilon(t) + M^\dagger(p,t,q^{-1})y(t), \quad (7)$$

where $M^\dagger(p,t,q^{-1})$ denotes the left inverse of $M(p,t,q^{-1})$, Problem (6) can be reformulated as indicated in (8), where the argument q^{-1} has been dropped for the sake of space.

Here comes the second fact as follows. *Since the only signals appearing in* (8) *are u, y and p, $\mathcal{D}_N$ can be used instead of the dynamic system relation as indicated by the first constraint of* (8). The problem can then be rewritten as illustrated in (9) where u, y and p come from the available dataset $\{u(t), y(t), p(t)\}_{t=1}^N$. Notice that in the above formulation:

- Problem (9) is independent of the analytical description of $A(q^{-1},p)$ and $B(q^{-1},p)$ and therefore no model identification is needed.
- The information about the data generation mechanism is implicitly included in $\mathcal{D}_N$.
- Problem (9) is generally nonconvex because of the product between the optimization variables ε and the parameters θ characterizing $B_K(q^{-1},p,\theta)$. Specifically, it is convex only if $B_K(q^{-1},p,\theta)$ is independent of θ, whereas it is *bi-convex* in case of any linear dependance of $B_K(q^{-1},p,\theta)$ on θ.

It should be here mentioned that the computation of the inverse of the reference map is not straightforward. However, for reference maps given in the state-space form (2), the result of the following Proposition can be employed.

Proposition 1: Assume that $D_M(p,t) \neq 0, \forall p$ in (2) such that $\exists\ D_M^{-1}(p,t)$ with $D_M^{-1}(p,t)D_M(p,t) = 1,\ \forall p$. Define the state-space representation of the inverse map of (2) as

$$\begin{aligned} x_{M^\dagger}(t+1) &= A_{M^\dagger}(p,t)x_{M^\dagger}(t) + B_{M^\dagger}(p,t)y(t) \\ r(t) &= C_{M^\dagger}(p,t)x_{M^\dagger}(t) + D_{M^\dagger}(p,t)y(t). \end{aligned} \quad (10)$$

The system matrices in (10) can be computed from $A_M(p,t)$, $B_M(p,t)$, $C_M(p,t)$ and $D_M(p,t)$ as follows:

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

Proof: See [6]. ■

Remark 4: In case of $D_M = 0$, to compute the inverse, an approximation of $D_M = \epsilon_D$, where $\epsilon_D << 1$, can be used, as it is common in robust control. Another more practical way to overcome the problem will be shown in Section VI. ■

Notice that if the data is noiseless and **A1** holds, (9) is the same as (6) and their minimizer yields $\varepsilon = 0$. This is clearly not the case for $w \neq 0$, since only y_w (and not y) could be used in (9) to replace (1), leading to a bias of the estimate. This situation will be dealt with in the next section.

IV. LPV CONTROLLER TUNING FROM DATA: NOISY DATA

In this Section, Problem 2 will be addressed and therefore, from now on, y_w in (5) will be considered as the available output signal in the data set. To deal with noisy data, the controller parameters θ will be now estimated on the basis of the *instrumental variable* (IV) scheme described in the sequel. It will be shown that this IV approach not only provides an extension to the noisy case, but also transforms the bi-convex optimization into a convex scheme.

Define the regressors $\phi(\xi,t)$ and $\phi(\tilde{\xi},t)$ according to (12), where the definitions of signals $\tilde{\xi}(t)$ and $\xi(t)$ are

$$\tilde{\xi}(t) = M^\dagger(p,t)y(t) - y(t),\ \ \xi(t) = M^\dagger(p,t)y_w(t) - y_w(t). \quad (11)$$

$$
\begin{array}{lrcll}
\min_{\theta,\varepsilon} \|\varepsilon\|^2 & & & & \\
\text{s.t.} & \varepsilon(t) & = & M(p,t,q^{-1})r(t) - y(t), & \forall t \in \mathcal{I}_1^N, \\
& A(p,t,q^{-1})y(t) & = & B(p,q^{-1})u(t), & \forall t \in \mathcal{I}_1^N, \\
& A_K(q^{-1},p,\theta)u(t) & = & B_K(q^{-1},p,\theta)(r(t)-y(t)), & \forall t \in \mathcal{I}_1^N.
\end{array} \tag{6}
$$

$$
\begin{array}{lrcll}
\min_{\theta,\varepsilon} \|\varepsilon\|^2 & & & & \\
\text{s.t.} & A(p,t)y(t) & = & B(p,t)u(t), & \forall t \in \mathcal{I}_1^N, \\
& A_K(p,t,\theta)u(t) & = & B_K(p,t,\theta)(M^\dagger(p,t)\varepsilon(t) + M^\dagger(p,t)y(t) - y(t)), & \forall t \in \mathcal{I}_1^N.
\end{array} \tag{8}
$$

$$
\begin{array}{lrcll}
\min_{\theta,\varepsilon} \|\varepsilon\|^2 & & & & \\
\text{s.t.} & A_K(p,t,\theta)u(t) & = & B_K(p,t,\theta)(M^\dagger(p,t)\varepsilon(t) + M^\dagger(p,t)y(t) - y(t)), & \forall t \in \mathcal{I}_1^N.
\end{array} \tag{9}
$$

$$
\begin{array}{llll}
\phi(\xi,t) = [& -u(t-1)f_{1,0}(p,t) & -u(t-1)f_{1,1}(p,t)\dots & -u(t-1)f_{1,n_0}(p,t)\dots \\
& -u(t-n_{\mathrm{a}_K})f_{n_{\mathrm{a}_K},0}(p,t) & -u(t-n_{\mathrm{a}_K})f_{n_{\mathrm{a}_K},1}(p,t)\dots & -u(t-n_{\mathrm{a}_K})f_{n_{\mathrm{a}_K},n_{n_{\mathrm{a}_K}}}(p,t) \\
& \xi(t)g_{0,0}(p,t) & \xi(t)g_{0,1}(p,t)\dots & \xi(t)g_{0,m_1}(p,t)\dots \\
& \xi(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},0}(p,t) & \xi(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},1}(p,t)\dots & \xi(t-n_{\mathrm{b}_K})g_{n_{\mathrm{b}_K},m_{n_{\mathrm{b}_K}}}(p,t)\ ]^\top
\end{array} \tag{12}
$$

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

$$
u(t) = \phi^\top(\xi,t)\theta + B_K(p,t,\theta)M^\dagger(p,t)\varepsilon(t). \tag{13}
$$

Consider now the optimization problem

$$
\hat{\theta}_{\mathrm{IV}} = \arg\min_\theta \sum_{t=1}^{N} \left\|\zeta(t)\left(\phi^\top(\xi,t)\theta - u(t)\right)\right\|_2^2, \tag{14}
$$

where $\zeta(t)$ is the *instrument*, a vector that shares the same dimension of $\phi(\xi,t)$ and is chosen by the user so that $\zeta(t)$ is independent of the noise term $\xi(t) - \tilde{\xi}(t) = (M^\dagger(p,t) - 1)w(t)$, *i.e.*, since $\mathbb{E}[w(t)] = 0$,

$$
\mathbb{E}[\zeta(t)(M^\dagger(p,t) - 1)w(t)] = 0,\ \forall t \in \mathcal{I}_1^N. \tag{15}
$$

Notice that (14) is aimed to minimize the coloured residual $B_K(p,t,\theta)M^\dagger(p,t)\varepsilon(t)$ in (13). By introducing the matrix notation

$$
Z = [\zeta(1)\ \dots\ \zeta(N)]^\top,\ U = [u(1)\ \dots\ u(N)]^\top,
$$

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

Problem (14) can be also written in the compact form

$$
\hat{\theta}_{\mathrm{IV}} = \arg\min_\theta \left\|Z^\top(\Phi\theta - U)\right\|_2^2, \tag{16}
$$

whose solution is given by

$$
\hat{\theta}_{\mathrm{IV}} = \left(Z^\top\Phi\right)^{-1} Z^\top U. \tag{17}
$$

Notice that (17) only depends on the data and $M^\dagger$, whereas no information about the structure of G or the noise model is required, analogously to the solution of (6). The following result shows that the solution of (17) asymptotically converges to the solution of problem (6), even if the data in (17) is noisy.

Proposition 2: The controller parameters $\hat{\theta}_{\mathrm{IV}}$ in (17) asymptotically converge *with probability 1* (w.p. 1) to the optimal parameters $\hat{\theta}$ in (6), that is

$$
\lim_{N\to\infty} \hat{\theta}_{\mathrm{IV}} = \hat{\theta}. \tag{18}
$$

Proof: See [6]. ■

There are several ways to build the instrument. One practical solution is to perform a second experiment on the plant, with the same input and parameter trajectories, and collect the output measurements now characterized by a different realization of noise (thus independent of that of the first experiment). A more efficient way to deal with this issue would be to resort to *Refined Instrumental Variables* (RIV) [9].

V. COMPARISON WITH 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, non-iterative methods, *i.e.* VRFT [4] and Non-iterative CbT [14], are considered here.

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

$$
\hat{\theta}_{\mathrm{vr}} = \arg\min_{\theta,\varepsilon_u} \|\varepsilon_u\|^2, \tag{19}
$$

where

$$
\varepsilon_u(t) = u(t) - K(q^{-1},\theta)e_v(t),\ t = 1,\dots,N
$$

$K(q^{-1},\theta)$ is the controller transfer function, $e_v(t) = M^{-1}(q^{-1})y(t) - y(t)$ and M represents an 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_v is the difference between the "virtual" reference signal feeding the closed-loop system

$$r_v(t) = M^{-1}(q^{-1})y(t), \tag{20}$$

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

- minimizing $\|\varepsilon_u\|^2$ is not the same as minimizing $\|\varepsilon\|^2$ in the original model-matching problem, as indicated by the authors of [4] themselves;
- even though also 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 computed in VRFT (20).
- 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 [7],

- 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 mapping needs to be LTI.

Regarding Non-iterative CbT, the differences with the proposed approach are even more evident. First of all, as indicated in [8], CbT cannot be extended to nonlinear systems, since the tuning scheme is based on the commutation of the plant and the controller. Moreover, the treatment of noise is based on extended instrumental variables minimizing a measure of the correlation between u and ε.

It should be also 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 will be an objective of future work.

VI. NUMERICAL EXAMPLE

In this Section, the effectiveness of the proposed approach is demonstrated on a numerical example. Consider the SISO LPV system G defined as

$$\begin{aligned} x_G(t+1) &= p(t)x_G(t) + u(t) \\ y(t) &= x_G(t), \end{aligned} \tag{21}$$

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

Let the desired behavior for the closed-loop system M be given by the second order plant

$$\begin{aligned} x_M(t+1) &= A_M(p,t)x_M(t) + B_M(p,t)r(t) \\ y_M(t) &= C_M(p,t)x_M(t) + D_M(p,t)r(t). \end{aligned} \tag{22}$$

where

$$A_M(p,t) = \begin{bmatrix} -1 & 1 \\ -1-\Delta p(t) & 1 \end{bmatrix}, \ B_M(p,t) = \begin{bmatrix} 1+p(t) \\ 1+\Delta p(t) \end{bmatrix},$$

$$C_M = [1 \ \ 0], \ D_M = 0,$$

$$\Delta p(t) = p(t) - p(t-1)$$

and y_M is the desired closed-loop trajectory for $y(t)$. Such a control objective has been selected for this example as the closed-loop matrices are easily computable and **A1** holds. *In practice, any reference model is allowed.*

Assume now that a gain-scheduled PI controller K 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

$$\theta_0(p,t) = \theta_{00} + \theta_{01}p(t), \tag{23}$$

$$\theta_1(p,t) = \theta_{10} + \theta_{11}p(t-1), \tag{24}$$

is available for model reference control of G. The closed-loop dynamics can be written as a function of the controller parameters as:

$$\begin{aligned} x_F(t+1) &= A_F(p,t)x_F(t) + B_F(p,t)r(t), \\ y(t) &= C_F(p,t)x_F(t) + D_F(p,t)r(t). \end{aligned} \tag{25}$$

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 is evident that there exists a controller in the considered class which is able to achieve a closed-loop behavior equal to M, *i.e.*, **A1** holds. Specifically, the parameters of the optimal controller are such that

$$\theta_0^\circ(p,t) = \theta_{00}^\circ + \theta_{01}^\circ p(t) = 1 + p(t), \tag{26}$$

$$\theta_1^\circ(p,t) = \theta_{10}^\circ + \theta_{11}^\circ p(t-1) = -p(t-1). \tag{27}$$

In this example, the parameters of the controller will be computed using the method proposed in this paper, without deriving a model of G, so as to design the control law directly from data in its IO form

$$\begin{aligned} u(t) = &u(t-1) + \theta_0(p,t)\,(r(t) - y(t)) + \\ &+ \theta_1(p,t)\,(r(t-1) - y(t-1)). \end{aligned} \tag{28}$$

For this purpose, a data set $\mathcal{D}_N$ of $N = 1000$ measurements are collected, by performing an experiment where $u(t)$

is selected as a white noise sequence with uniform distribution $\mathcal{U}(-1,1)$ and $p(t) = 0.4\sin(0.04\pi t)$. The output measurements are corrupted by a white noise sequence 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.

As a preliminary step, recall that $M^\dagger$ is needed to compute (11). 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. Consider the system M' defined as

$$\begin{aligned} x_{M'}(t+1) &= A_{M'}(p,t)x_{M'}(t) + B_{M'}(p,t)r(t) \\ y_{M'}(t) &= C_{M'}(p,t)x_{M'}(t) + D_{M'}(p,t)r(t) \end{aligned}$$

where $A_{M'}(p,t) = A_M(p,t)$, $B_{M'}(p,t) = B_M(p,t)$, $C_{M'}(p,t) = C_M(qp)A_M(p,t)$ and $D_{M'}(p,t) = C_M(qp)B_M(p,t)$. Notice that, to compute $C_{M'}(p,t)$ and $D_{M'}(p,t)$, the sequence of p in C_M has to be shifted one step forward in time. It is then easy to check that $y_{M'}(t) = y_M(t+1)$. Since now $D_{M'} \neq 0\ \forall p \in \mathbb{P}$, the inverse $M'^\dagger$ of M' can be computed using Proposition 1 and, as a consequence, (11) is given by filtering y_w with $M'^\dagger$ and shifting the data in time as

$$\tilde{\xi}(t) = M'^\dagger(p,t)y_w(t+1) - y_w(t),\ t \in \mathcal{I}_1^{N-1}. \tag{29}$$

It should be underlined here that, doing so, the samples available for controller identification become $N-1$. This procedure is feasible because filtering is operated off-line.

The controller parameters can now be computed using the direct data-driven method proposed in this paper, *i.e.*, the IV estimation formula (17) using an instrument built with a second experiment. The resulting values are

$$\theta_0(p,t) = 0.9852 + 1.0166p(t), \tag{30}$$
$$\theta_1(p,t) = -0.0153 - 0.9860p(t-1), \tag{31}$$

where small discrepancies with respect to (26)-(27) are obviously due to the noise and the finiteness of N.

Despite these small variations, the controller appears to be effective in terms of matching of the desired closed-loop behavior. As an example, Figure 2 illustrates a reference tracking (validation) experiment using a piecewise linear p, different from the trajectory of the estimation dataset $\mathcal{D}_N$. Notice that, in these simulations, the mean tracking error is less than 1%.

VII. CONCLUSIONS AND FUTURE WORKS

In this paper, a novel data-driven method has been introduced to design model-reference controllers for LPV systems using a set of IO data without undertaking a full modeling study. An instrumental variable technique has been proposed to deal with the bi-convex optimization issue related to the given problem formulation and the effectiveness of the approach has been proven on a numerical example. This paper aims to lay the basic foundations for future research in direct data-driven control of LPV plants. Specifically, future activities will be devoted to the development of:

- control design techniques using collection of closed-loop data (useful to deal with unstable plants);
- formal methods for controller structure selection;
- optimization methods including stability constraints.

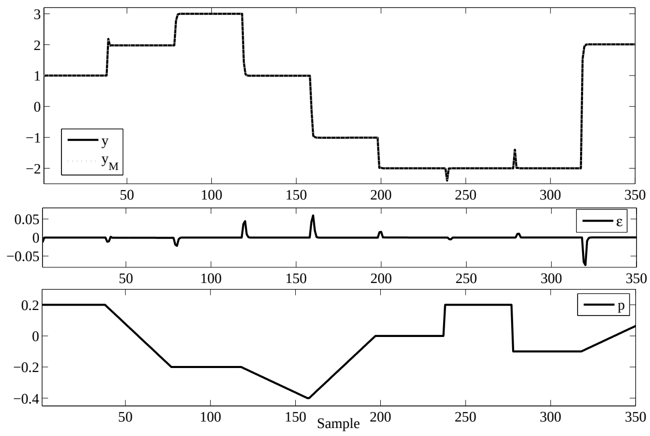

Fig. 2. Numerical example: realized closed-loop output response y and desired output response y_M (top), tracking error ε (middle), scheduling signal p (bottom).

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 (GA), USA*, pages 4018–4023, 2010.

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

[3] 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.

[4] 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.

[5] 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*, 2012.

[6] S. Formentin, D. Piga, R. Tóth, and S. Savaresi. LPV control system design from data. *Technical report TUE-CS-2013-004, Department of Electrical Engineering, TU Eindhoven*, 2013.

[7] S. Formentin and S. Savaresi. Virtual reference feedback tuning for linear parameter-varying systems. In *IFAC World Congress*, volume 18, pages 10219–10224, 2011.

[8] 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.

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

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

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

[12] 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.

[13] 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.

[14] 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.

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