Received: 22 April 2019 | Revised: 23 October 2019 | Accepted: 28 October 2019

DOI: 10.1002/rnc.4821

SPECIAL ISSUE ARTICLE

WILEY

Direct data-driven design of switching controllers

Valentina Breschi | Simone Formentin

Dipartimento di Elettronica, Bioingegneria e Informazione, Politecnico di Milano, Milan, Italy

Correspondence
Valentina Breschi, Dipartimento di Elettronica, Bioingegneria e Informazione, Politecnico di Milano, Via G. Ponzio 34/5, 20133 Milan, Italy.
Email: valentina.breschi@polimi.it

Funding information
Fondazione Cariplo, Grant/Award Number: 2017-1520

Summary

Switching linear models can be used to represent the behavior of hybrid, time-varying, and nonlinear systems, while generally providing a satisfactory trade-off between accuracy and complexity. Although several control design techniques are available for such models, the effect of modeling errors on the closed-loop performance has not been formally evaluated yet. In this paper, a data-driven synthesis scheme is thus introduced to design optimal switching controllers directly from data, without needing a model of the plant. In particular, the theory will be developed for piecewise affine controllers, which have proven to be effective in many real-world engineering applications. The performance of the proposed approach is illustrated on some benchmark simulation case studies.

KEYWORDS

data-driven control, model-free control, piecewise affine systems, switching systems

1 | INTRODUCTION

Switching models are well-established powerful tools that can be used to describe hybrid, nonlinear, and time-varying phenomena occurring in modern engineering applications, eg, when a system may work in different operating conditions or the dynamics may vary according to specific physical limits or switches.[1] In particular, piecewise affine (PWA) models are collections of linear time-invariant (LTI) submodels, defined over nonoverlapping polyhedral regions of the input+state domain and indexed by a discrete-valued *switching variable*. PWA models have been extensively used and studied within the control community for their universal approximation properties[2] that make them suited for the analysis and control of several classes of nonlinear systems,[3] and for their equivalence with other classes of hybrid models.[4] The advantages of working with such model structures are many, but most importantly, they provide an excellent trade-off between descriptive power and complexity (the local submodels may be very simple even if the overall system is highly complex).

In the last decades, several control methods have been proposed to handle systems characterized by the interaction between continuous and discrete dynamics, eg, based on Lyapunov arguments[5] or on the solution of linear matrix inequalities (LMI)[6-8] (see the work of De Schutter et al[9] for an overview on existing approaches). Under additional operating constraints, PWA systems can be effectively controlled via *hybrid model predictive control*.[10,11] However, based on the model dynamics, the controller may turn out to be highly complex and of high-order, leading to many robustness and implementation problems.[12] Moreover, hardware computational limits may prevent or restrain the use of MPC or optimization-based solutions. Fixed-structure control design, eg, switching proportional integral derivative (PID) control laws with PWA variations of the gains, could instead represent an effective alternative approach for all the above cases. Unfortunately, unlike the LTI case,[13] little work has been done on fixed-order control design for switching systems, eg, in the work of Kong et al,[14] a switching PID is tuned based on an accurate model of the system using genetic algorithms to control a nonlinear DC motor. The switching logic is instead dictated by ad hoc heuristics in the work of Marchetti et al.[15]

In the general case where no accurate models are available, the design of a fixed-order switching controller is a tough undertaking, in that it requires three critical design steps: (i) the parameterization and identification from experimental data of a switching model of the nonlinear/time-varying/hybrid system, (ii) a model-reduction step to compute a model of the desired complexity, and (iii) a control design step. While the latter is indeed a critical but feasible task, the first two steps offer several issues. First of all, the problem of identifying PWA models is known to be NP-hard.[16] For this reason, several heuristics[17,18] have been introduced to learn both the parameters of each affine submodel and the polyhedral partition of the input+state domain. Among them, the approaches proposed in[19-23] initially tackle the problem of identifying the local models and, afterwards, the one of learning the polyhedral partition. Among these techniques, only the one proposed in[23] can efficiently handle large data sets, thanks to the computationally efficient approach proposed to solve the multicategory discrimination problem. Secondly, model reduction for switching systems is a largely unexplored world. Gosea et al[24] rely on the assumption that the switching signal is known. The same holds for Shaker and Wisniewski,[25] which also requires the solution of a set of LMIs, thus making the method computationally expensive. The work of Papadopoulos and Prandini[26] presents an interesting technique for the case of endogenous switching signal; however, such a method is conceived for continuous time systems, whereas most of existing identification algorithms compute models in discrete time.[18,27] It follows that the continuous time counterpart of the model would need to be found, eg, as explained in[28] or, alternatively, the system needs to be directly identified in continuous-time as described in.[29] Nevertheless, notice that, in[29] the switching sequence is supposed to be known.

Regardless of the specific technique for identification or model-reduction, the above control design approach strongly relies on the availability of an accurate model of the system. Since the controller complexity is specified *a priori*, an appealing alternative is the *direct optimization from data* of the controller parameters, without undertaking an expensive and time-consuming modeling study or performing any model-reduction step.[30] Such an approach to control design is usually referred to as *direct data-driven control* or *model-free control* and has been widely investigated within the LTI framework, see, eg, the *Virtual Reference Feedback Tuning* (VRFT)[31,32] and the *Non-iterative Correlation-based Tuning* (CbT)[33,34] methods. Although such approaches have been extended to the control of more complex systems, such as *linear parameter-varying* (LPV) systems[35,36] or nonlinear systems with bounded disturbances,[37,38] they have never been extended to the control of hybrid/switching systems, in which both the local behavior and the operating condition of the plant are *unknown*. To the best of our knowledge, the only approach in the literature for data-driven control of switched systems has been recently presented in,[39] in which the problem of finding a switching static regulator for state feedback is solved within a set-membership framework through the solution of a polynomial optimization problem. This approach relies on Lyapunov arguments to formulate the design problem and to prove the stability of the closed-loop system. This is possible since the authors assume the state of the plant to be measurable, which is often not true in practice. Although a state vector can be constructed relying on past input/output data, properties proven by using this non-minimal state might not extend to the actual system. The authors further assume the discrete state of the controlled system to be known, implying insights on the true system and limiting the use of the approaches when the actual plant is only approximated by a switching model.

Similar assumptions are considered in most of existing approaches for *model reference adaptive control* (MRAC) for switching and PWA systems, so that closed-loop stability can be guaranteed through Lyapunov arguments. Since the first contributions on the extension of adaptive control to this class of hybrid systems in,[40,41] MRAC methods for PWA systems rely on the assumption that the partition driving the switches in the operating mode of the system is known. This limits the application of the method, especially when dealing with unknown nonlinear systems. Although the approaches presented in[41] have been further extended in[42] to overcome possible instabilities caused by disturbances on the input, the effect of uncertain/noisy state measurements has not been explored. Analogously, the hypothesis of exact state knowledge is shared by the approach proposed in[43] for asynchronous adaptive tracking control for switched system, where the authors further assume prior knowledge on the lower and upper bounds on the controller parameters. More recently, two MRAC approaches for PWA systems have been proposed in,[44] which rely on different strategies to adaptively tune the controller's parameters. Indeed, in the *direct* method presented therein, parameter tuning is based on the tracking error, whereas the *indirect* approach relies on a time-varying estimate of the model of the system. Although the approaches handle multivariable plants and are not constrained by the same structural assumptions on the plant imposed in,[40,41] both methods still assume the operating condition of the system to be known. Furthermore, the theoretical analysis of the methods does not consider disturbances affecting the state measurements, which is instead introduced in the numerical examples presented to assess the performance of the controller. An approach for adaptive control of uncertain switched systems has been proposed in,[45] which accounts for disturbances acting on the measured state and does not require prior knowledge on the bounds of the uncertainty set. However, as in most approaches for control of switched systems, the operating condition of the plant is supposed to be directly accessible, so that it can be imposed by the controller. More recently an MRAC approach for switched systems explicitly accounting for the case of unmeasurable state has been proposed in.[46] The problem is handled by designing both a state observer and an adaptive controller, which requires quite an extensive design effort with respect to the one needed to design the controller only. Furthermore, both the controller and the observer relies on the discrete state of the plant, which is assumed to be accessible.

Instead of adjusting the controller to changes in the operating condition of the plant online, like in adaptive approaches,[47] in this paper we present a novel model-reference approach for the direct data-driven control of PWA systems, focusing on the *single-input–single-output* (SISO) setting only. This allows us to set the foundations for future extensions to the *multiple-input–multiple-output* (MIMO) case, in which not only the nonlinear nature of the plant has to be accounted for but also (unknown) input couplings. The proposed method allows for the offline design of fixed-order PWA controllers from data within a stochastic setting, without requiring the modeling/identification of the *unknown* plant. The controller is thus characterized by a finite collection of local affine controllers defined over a polyhedral partition of a fixed domain, which are not adjusted/modified once the controller has been deployed. The proposed technique relies on *input-output* (IO) data only, so that the state of the system to be controlled does not need to be measured. Additionally, we assume that the operating condition of the controlled system is *unknown*. Thanks to this assumption and given the universal approximation properties of PWA maps,[2] the proposed technique is suited also for the control of nonlinear time-varying systems, which can often be well approximated with PWA models. Inspired by recent results on model-reference direct control of LPV systems,[48] the design problem is formulated as a convex optimization problem, whose solution allows us to characterize both the discrete and the continuous dynamics of the controller. This optimization problem is solved through an extension of the *coordinate descent method* for learning jump models proposed in,[49] which embeds an *instrumental variable* (IV) scheme to efficiently cope with the noise affecting the measurements of the system output. Because of the chosen formulation, the partition characterizing the discrete dynamics of the controller is retrieved from data, by using the computationally efficient multicategory discrimination approach proposed in.[23]

The main contributions of the paper are the following: (i) a novel data-driven stochastic method for the estimation of fixed-order PWA controller is proposed, which does require neither the knowledge of a model for the underlying system nor that of its operating condition; (ii) through the estimation of both local controllers and the switching mechanism from data, the method represents an alternative to existing data-based approaches without requiring the prior design of the switching policy, which is instead directly estimated from offline data; (iii) to the best of our knowledge, this is the first approach that allows for the direct design of both local controllers and switching policy; (iv) thanks to the estimation of the switching policy, we show that the method can effectively be used for the direct control of nonlinear systems, whereas other approaches for data-driven control of switching systems cannot be employed like that, since they rely on the knowledge of the system switching sequence. Despite these advantages, currently closed-loop stability cannot be formally guaranteed, due to the lack of off-the-shelf data-driven conditions for closed-loop stability within the input-output switching setting. Nonetheless, if the chosen reference model is stable and the problem is correctly solved, closed-loop stability is expected to be attained in practice since a stable behavior is matched. The choice of an optimal attainable reference model is thus crucial to achieve good closed-loop performance. Although outside the scope of this work, a way of handling this interesting and challenging problem can be to extend the approach proposed in the work of Selvi et al[50] to the switching case.

The paper is organized as follows. The problem is formally stated in Section 2. Section 3 is devoted to the reformulation of the objective in a *direct data-driven* fashion. The proposed design method is described in Section 4, whereas the details on the required preprocessing phase are given in Section 5. In Section 6, the effectiveness of the approach is assessed on two numerical example and a benchmark simulation case study. This paper is ended by some concluding remarks.

Notation

Let $\mathbb{N}$ be the set of natural numbers and $\mathbb{Z}$ be the set of integer numbers. Denote with $\mathbb{R}$ and $\mathbb{R}^+$ the sets of real and positive real numbers, respectively. Let $\mathbb{R}^n$ be the set of n-dimensional real vectors and denote with $\mathbb{R}^{n\times m}$ the set of real matrices of dimension $n \times m$. Denote with $\mathbf{1}_n \in \mathbb{R}^n$ the n-dimensional unitary vector. Given a vector $a \in \mathbb{R}^n$, its i-th component is denoted as a_i, $\|a\|_2$ is its Euclidean norm and $(a)_+$ is a vector whose i-th element is $\max\{0, a_i\}$. Given two vectors $a, b \in \mathbb{R}^n$, $\max\{a, b\}$ is the vector whose i-th component is $\max\{a_i, b_i\}$, and let $a \preceq b$ denote the component-wise inequality $a_i \leq b_i$, for $i = 1, \ldots, n$. Given a matrix $A \in \mathbb{R}^{n\times m}$, $A' \in \mathbb{R}^{m\times n}$ is its transpose. Given $\{u_t\}_{t\geq 0}$ with $t \in \mathbb{N}$, the shift operator q^{-1} is such that for $d \in \mathbb{Z}$ it holds $q^{-d}u_t = u_{t-d}$. Given a logic condition C, denote with $\mathbb{1}_{[C]}$ the corresponding indicator function, ie,

$$\mathbb{1}_{[C]} = \begin{cases} 1 & \text{if} \quad C = \text{true}, \\ 0 & \text{otherwise}. \end{cases} \tag{1}$$

For example, for $a, b \in \mathbb{R}$, $\mathbb{1}_{[a=b]}, = 1$ iff $a = b$.

2 | PROBLEM FORMULATION

Let $\mathcal{G}_s$ be a single-input single-output (*SISO*) *switching* system that can operate according to $K \in \mathbb{N}$ finite operating regimes (also denoted as *modes*). The input-output (*IO*) relationship characterizing $\mathcal{G}_s$ is described by the difference equation

$$A(s_o(t), q^{-1})y_o(t) = B(s_o(t), q^{-1})u(t) + f(s_o(t)), \tag{2a}$$

where $u(t) \in \mathbb{R}$ and $y_o(t) \in \mathbb{R}$ are the input to $\mathcal{G}_s$ and the corresponding noiseless output, respectively, and the *switching variable* $s_o(t) \in \{1, \dots, K\}$ indicates the operating mode of the system at time $t \in \mathbb{N}$. The scalar $f(k) \in \mathbb{R}$ and the polynomials $A(k, q^{-1})$ and $B(k, q^{-1})$ in the shift operator q^{-1} characterize the dynamics of $\mathcal{G}_s$ at the k-th mode, for $k = 1, \dots, K$, with $A(k, q^{-1})$ and $B(k, q^{-1})$ respectively defined as

$$A(k, q^{-1}) = 1 + \sum_{i=1}^{n_a} a_i(k)q^{-i}, \tag{2b}$$

$$B(k, q^{-1}) = \sum_{i=1}^{n_b} b_i(k)q^{-i}, \tag{2c}$$

and $n_a, n_b \in \mathbb{N}$ indicating the dynamical order of the system.

Let $X_o(t)$ be the collection of past inputs/outputs of the system given by

$$X_o(t) = \begin{bmatrix} y_o(t-1), & \dots & y_o(t-n_a), & u(t-1), & \dots & u(t-n_b) \end{bmatrix}' \in \mathcal{X} \subseteq \mathbb{R}^{n_x}. \tag{3}$$

In this work, we focus on picewise affine systems (*PWA*) systems. Indeed, we assume that $\mathcal{G}_s$ changes its operating regime according to a *polyhedral partition* of the domain $\mathcal{X}$. Therefore, we assume that there exists a complete partition $\{\mathcal{X}_k\}_{k=1}^K$ of space $\mathcal{X}$ such that the *discrete dynamics* of the switching system $\mathcal{G}_s$ is described by the logical condition

$$s_o(t) = k \quad \iff \quad X_o(t) \in \mathcal{X}_k, \quad t \in \mathbb{N}. \tag{4}$$

Note that the collection $\{\mathcal{X}_k\}_{k=1}^K$ forms a complete partition of space $\mathcal{X}$ if $\bigcup_{k=1}^K \mathcal{X}_k = \mathcal{X}$ and $\overset{\circ}{\mathcal{X}}_i \cap \overset{\circ}{\mathcal{X}}_j = \emptyset, \forall i \neq j$, with $\overset{\circ}{\mathcal{X}}_i$ indicating the interior of $\mathcal{X}_i$.

The convex polyhedral region $\mathcal{X}_k$ is defined as

$$\mathcal{X}_k = \{X_o(t), \quad t \in \mathbb{N} : \mathcal{H}_k X_o(t) \preceq \mathcal{D}_k\}, \tag{5}$$

with $\mathcal{H}_k \in \mathbb{R}^{\mu_k \times n_x}$, $\mathcal{D}_k \in \mathbb{R}^{\mu_k}$ and μ_k being the number of linear inequalities defining the k-th polyhedron, for $k \in \{1, \dots, K\}$.

Example 1. Consider the linear switching system $\mathcal{G}_s$ in state-space form

$$x(t+1) = 0.8 \begin{bmatrix} \cos\alpha(t) & -\sin\alpha(t) \\ \sin\alpha(t) & \cos\alpha(t) \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), \tag{6a}$$

$$y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} x(t), \tag{6b}$$

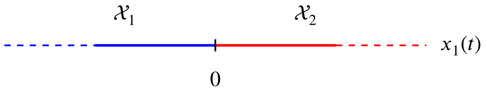

FIGURE 1 Polyhedral partition of the space driving the change in the operating condition of the system in (6) [Colour figure can be viewed at wileyonlinelibrary.com]

where

$$\alpha(t) = \begin{cases} \frac{\pi}{3} & \text{if} \quad [\,1 \;\; 0\,]x(t) \geq 0, \\ -\frac{\pi}{3} & \text{if} \quad [\,1 \;\; 0\,]x(t) < 0. \end{cases} \tag{6c}$$

It is straightforward to see that the operating condition of the system in (6) is driven by the polyhedral partition of the output space reported in Figure 1. This system is indeed piecewise linear (PWL).

In this paper, we aim at designing a controller $\mathcal{C}_s$ for $\mathcal{G}_s$ to match a desired stable closed-loop behavior $\mathcal{M}_s$. The considered reference-models lie within the class of *PWL* models, described by the difference equations

$$x_M(t+1) = A_M(s_M(t))x_M(t) + B_M(s_M(t))r(t), \tag{7a}$$

$$y_d(t) = C_M(s_M(t))x_M(t) + D_M(s_M(t))r(t), \tag{7b}$$

with $x_M(t) \in \mathcal{X}^M \subseteq \mathbb{R}^{n_{x_M}}$ and $y_d(t) \in \mathbb{R}$ being the desired closed-loop output for the reference $r(t) \in \mathbb{R}$ at time t. The discrete variable $s_M(t) \in \{1, \dots, K_M\}$ changes its value according to a polyhedral partition of the state space $\mathcal{X}^M$, ie,

$$s_M(t) = i \iff x_M(t) \in \mathcal{X}_i^M, \;\; i \in \{1, \dots, K_M\}. \tag{7c}$$

According to the switching logic described in (7c), the operating condition of the reference model varies independently of the mode of $\mathcal{G}_s$ and the reference signal. This choice allows us to design the controller $\mathcal{C}_s$, without restricting its effectiveness to a specific class of references. With a slight abuse of notation, the operator $M(s_M(t))$ is used from now on to denote the relationship between $r(t)$ and $y_d(t)$, namely, $y_d(t) = M(s_M(t))r(t)$.

Throughout the rest of this paper, the system $\mathcal{G}_s$ is assumed to be fully unknown. Nonetheless, we suppose that open-loop experiments can be carried out by feeding $\mathcal{G}_s$ with a set of inputs $\{u(t)\}_{t=1}^T$ and that the corresponding output can be measured. It is worth remarking that the operating condition of $\mathcal{G}_s$ is *unknown* and it cannot be retrieved from data without identifying a model for the plant. For the collected signals to be suited for learning tasks, the PWA system $\mathcal{G}_s$ is assumed to be *stable*, where stability is intended as follows:

Definition 1. A PWA system (2)+(4) is *bounded-input–bounded output* (BIBO) stable if, for all trajectories $\{u(t), y_o(t)\}_{t\in\mathbb{N}}$ verifying (2) and switching paths $\{s_o(t), \;\; t \in \mathbb{N} : s_t \in \{1, \dots, K\}, \quad \text{with} \quad K < \infty\}$ satisfying (4), it holds

$$\sup_{t\geq 0} |u(t)| < +\infty \;\; \Rightarrow \;\; \sup_{t\geq 0} |y_o(t)| < +\infty. \tag{8}$$

Note that this condition can be relaxed by collecting experimental data in closed-loop, through the interconnection of $\mathcal{G}_s$ with a stabilizing controller.[51] The output measurements are supposed to be corrupted by an additive zero-mean stationary noise $v(t)$, ie,

$$y(t) = y_o(t) + v(t), \qquad t = 1, \dots, T. \tag{9}$$

Instead of identifying $\mathcal{G}_s$, we propose to use the available data set $\mathcal{D}_T = \{u(t), y(t)\}_{t=1}^T$ to directly design a controller $\mathcal{C}_s$ for the system, so to match the desired closed-loop behavior $\mathcal{M}_s$ described by (7). For this purpose, the controller is sought within the class of PWA models

$$A_c(s_c(t), q^{-1})u(t) = B_c(s_c(t), q^{-1})e(t) + f_c(s_c(t)), \tag{10a}$$

where $e(t) = r(t) - y(t)$ and $s_c(t) \in \{1, \dots, K_c\}$, with $K_c \in \mathbb{N}$ finite, are the tracking error and the logic state of the controller at time t, respectively. The polynomials $A_c(k, q^{-1})$ and $B_c(k, q^{-1})$ are defined as

$$A_c(k, q^{-1}) = 1 + \sum_{i=1}^{n_a^c} a_i^c(k)q^{-i}, \tag{10b}$$

$$B_c(k, q^{-1}) = \sum_{i=0}^{n_b^c} b_i^c(k)q^{-i}, \tag{10c}$$

with n_a^c and n_b^c denoting the dynamical order of the k-th local controller, so that the parameters to be estimated for each local controller are

$$\theta_k = \left[a_1^c(k), \ \dots \ a_{n_a^c}^c(k), \ b_0^c(k), \ b_1^c(k), \ \dots \ b_{n_b^c}^c(k), \ f_c(k) \right]' \in \mathbb{R}^{n_\theta}, \quad k = 1, \dots, K_c. \tag{10d}$$

We choose the discrete state $s_c(t) \in \{1, \dots, K_c\}$ of the controller to be driven by a polyhedral partition $\{\mathcal{X}_i^c\}_{i=1}^{K_c}$ of $\mathcal{X}^c$, ie,

$$s_c(t) = i \iff \chi(t) \in \mathcal{X}_i^c, \tag{10e}$$

with $\chi(t)$ being generally defined as the following collection of inputs/outputs

$$\chi(t) = \left[u(t-1), \ \dots \ u\left(t - n_a^c\right), \ y(t), \ y(t-1), \ \dots y\left(t - n_b^c\right) \right]' \in \mathcal{X}^c \subseteq \mathbb{R}^{n_\chi}. \tag{10f}$$

Remark 1. By imposing that the controller depends on the tracking error e, different weights on the reference and the output are not allowed. Nonetheless, this choice complies with the characteristics of many controllers used in practice for SISO systems, ie, PID controllers. At the same time, it allows us to limit the number of parameters to be learned, thus reducing the computational burden of the final algorithm. Finally, note that the structure of $\mathcal{C}_s$ can be easily modified to account separately for y and r. Indeed, this would entail a slight change in the expression for the dynamics of the local controllers, while it does not modify the vector χ.

Remark 2. Given the chosen switching logic for $\mathcal{C}_s$ (see (10e)-(10f)), the partition characterizing the controller $\mathcal{C}_s$ is directly related to the one of the plant. Nonetheless, as $\mathcal{G}_s$ is unknown, such a relationship cannot be explicitly determined and will be therefore numerically computed as illustrated next.

The available dataset $\mathcal{D}_T$ is assumed to be collected by feeding $\mathcal{G}_s$ with sufficiently rich inputs, ie, the input sequence should be persistently exciting for all operating conditions of interest and it should enable to periodically visit all these modes. A preliminary check on the properties of the input can be performed as described in,[52] where a more restrictive definition of persistence of excitation is considered. Nonetheless, since the design of experiment for switching systems is a difficult open problem and is out of the scope of this paper, further discussions on this topic will be postponed to future works.

Under this assumption, the problem addressed in the paper can be stated as follows.

Problem 1 (Data-driven design of switching controllers). Given a set of data set $\mathcal{D}_T = \{u(t), y(t)\}_{t=1}^T$ collected from open-loop experiments on $\mathcal{G}_s$, described as in (2), and a PWL reference closed-loop model $\mathcal{M}_s$ as in (7), find the *fixed-order* PWA controller with the structure in (10) that realizes the desired closed-loop behavior $\mathcal{M}_s$.

Remark 3. If $K = 1$, problem (12) reduces to the data-driven design of a controller for a linear time invariant (*LTI*) system. When the reference model is also LTI (ie, $K_M = 1$), the considered design problem can thus be effectively solved via existing techniques, such as the Virtual Reference Feedback Tuning (*VRFT*) method[31] or Non-iterative Correlation-based Tuning (*CbT*).[33] Moreover, if the operating condition $s_o(t)$ of $\mathcal{G}_s$ is measured, $\mathcal{G}_s$ can be seen as an LPV system with integer scheduling variable, and thus Problem (12) can be solved via the approach proposed in.[48]

Problem 1 requires to learn from data: (i) the number K_c of local controllers, (ii) the overall parameter matrix $\Theta = [\theta_1, \dots, \theta_{K_c}]$ and (iii) the polyhedral partition $\{\mathcal{X}_i^c\}_{i=1}^{K_c}$ of domain $\mathcal{X}^c$. Since both the parameters $\Theta \in \mathbb{R}^{n_\Theta}$ and the partition $\{\mathcal{X}_i^c\}_{i=1}^{K_c}$ are not fixed a priori, the PWA controller is expected to be flexible enough to attain the desired closed-loop behavior, even if the underlying system can only be approximately described through a PWA model. On the other hand, since we aim at designing fixed-order local controllers, their dynamical order (namely, n_a^c and n_b^c) is fixed a priori.

In this work, the number of local controllers K_c is also imposed by the user. Although K_c is generally unknown, it can be chosen by practitioners relying on prior information on the possible working conditions of the system. Alternatively, since the design problem is solved offline, the number of local controllers can be chosen through cross-validation, with an upper bound on K_c dictated by the maximum tolerable complexity of the controller. As a possible cross-validation strategy, the design problem can be solved over a subset of data for increasing values K_c, and the number of local controllers can then be selected as the one minimizing a properly chosen performance criterion. A possible choice for such a criterion is indicated in Section 6. Otherwise, any clustering algorithm (eg, K-means[53]) can be used to split the vectors $\chi(t)$ into

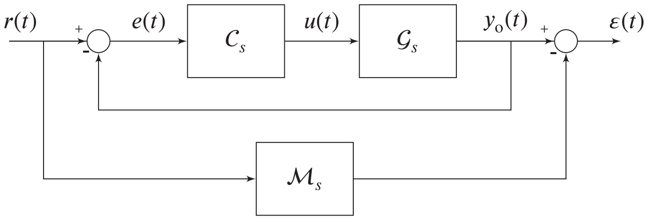

FIGURE 2 Proposed closed-loop behavior matching scheme

K_c clusters, for increasing values of K_c. The number of local models can then be chosen as the one leading to the *best* clustering solution according to some evaluation criterion, eg, the Davies-Bouldin index.[54]

Let us now mathematically formalize Problem 1. Define $\varepsilon(t)$ as the mismatch between the desired closed-loop behavior and the one attained interconnecting $\mathcal{C}_s$ and $\mathcal{G}_s$ at time t, ie,

$$\varepsilon(t) = y_o(t) - y_d(t), \tag{11}$$

as illustrated in Figure 2.

According to this matching scheme, we specifically search for a PWA controller $\mathcal{C}_s$, defined as in (10), that realizes the desired closed-loop behavior $\mathcal{M}_s$ in (7) with the least mismatch error $\varepsilon(t)$ in (11). Then, Problem 1 can be rewritten as the optimization problem

$$\underset{\varepsilon,\Theta,\{\mathcal{X}_i^c\}_{i=1}^{K_c},\mathcal{S}_c}{\text{minimize}} \ \|\varepsilon\|_{\ell_2}^2 \tag{12a}$$

$$s.t. \quad \varepsilon(t) = y_o(t) - M(s_M(t))r(t), \quad \forall t \in \mathcal{I}_1^T, \tag{12b}$$

$$A(s_o(t), q^{-1})y_o(t) = B(s_o(t), q^{-1})u(t) + f(s_o(t)), \quad \forall t \in \mathcal{I}_1^T, \tag{12c}$$

$$s_o(t) = k \iff X_o(t) \in \mathcal{X}_k, \quad k \in \{1, \dots, K\}, \quad \forall t \in \mathcal{I}_1^T, \tag{12d}$$

$$A_c(s_c(t), q^{-1})u(t) = B_c(s_c(t), q^{-1})(r(t) - y_o(t)) + f_c(s_c(t)), \quad \forall t \in \mathcal{I}_1^T, \tag{12e}$$

$$s_c(t) = i \iff \chi(t) \in \mathcal{X}_i^c, \quad i \in \{1, \dots, K_c\}, \quad \forall t \in \mathcal{I}_1^T, \tag{12f}$$

where $\mathcal{I}_1^T = \{1, 2, \dots, T\}$. The constraint in (12b) is formulated exploiting the desired closed-loop dynamics. Note that the cost function is optimized with respect to the unknown parameters Θ, the polyhedral partition $\{\mathcal{X}_i^c\}_{i=1}^{K_c}$, and the sequence of controller modes $\mathcal{S}_c = \{s_c(t)\}_{t=1}^T$. The introduction of this additional unknown is functional to the *clustering phase* needed to estimate both the parameters of the local controllers and the polyhedral partition of the space $\mathcal{X}^c$ in a supervised fashion. In the rest of this paper, for fixed K_c, n_a^c, and n_b^c, we assume that the following statements hold.

Assumption 1 (Perfect matching). There exists a parameter matrix Θ_o, a polyhedral partition $\{\mathcal{X}_{o,i}^c\}_{i=1}^{K_c}$ and an active mode sequence $\mathcal{S}_{o,c}$ such that the corresponding PWA controller $\mathcal{C}_s$ in (10) realizes $\mathcal{M}_s$ in closed-loop, ie, Θ_o, $\{\mathcal{X}_{o,i}^c\}_{i=1}^{K_c}$, $\mathcal{S}_{o,c}$ are the solution of problem (12).

Assumption 2 (Parameters identifiability). For fixed partition $\{\mathcal{X}_{o,i}^c\}_{i=1}^{K_c}$ and for every two instances $\Theta^{(1)}$ and $\Theta^{(2)}$ of the parameter matrix Θ, there exists an input trajectory such that the response of the interconnection between $\mathcal{C}_s$ in (10) and $\mathcal{G}_s$ in (2) is different if $\Theta^{(1)} \neq \Theta^{(2)}$.

Assumption 3 (Partition identifiability). For fixed parameters Θ_o and for every couple of partitions $\{\mathcal{X}_i^{c,(1)}\}_{i=1}^{K_c}$ and $\{\mathcal{X}_i^{c,(2)}\}_{i=1}^{K_c}$ of space $\mathcal{X}^c$, there exists an input trajectory such that the response of the interconnection between $\mathcal{C}_s$ in (10) and $\mathcal{G}_s$ in (2) is different if the partitions are different.

Assumption 1 entails that the considered problem is well-posed, namely, there exists a controller within the considered class $\mathcal{C}_s$ that realizes the desired closed-loop behavior. Instead, Assumptions 2 and 3 imply that the parameters and partition characterizing the *optimal* controller are globally identifiable, provided that the other unknowns are fixed to the optimum. These assumptions are required for consistency analysis, but they may be hard to check in practice. Indeed, the system $\mathcal{G}_s$ to be controlled is unknown and, thus, problem (12) cannot be explicitly solved. Nonetheless, it must be pointed out that these hypotheses do not limit the applicability of the approach since our main goal is not to exactly retrieve the optimal solution but to design a controller that behaves equivalently to the *optimal* solution to problem (12), namely, $\mathcal{C}_s$

leading to the desired closed-loop performance. Note that similar identifiability assumptions are shared by any control design method where the model is found from experimental data.

We further rely on the following technical assumption on $M(s_M(t))$.

Assumption 4. The mapping $M(s_M(t))$ between the reference signal $r(t)$ and the desired output $y_d(t)$ is invertible, for all $t \in \mathcal{I}_T$.

The left-inverse of $M(s_M(t))$ in (7) is denoted as $M^{\dagger}(s_M(t))$ and is defined as follows.

Definition 2. Given a causal PWA model $M(s_M(t))$ with input $r(t)$ and output $y_d(t)$, defined as in (7), its left-inverse is the causal PWA mapping $M^{\dagger}(s_M(t))$ giving $r(t)$ as an output when fed with $y_d(t)$, ie, $M^{\dagger}(s_M(t))M(s_M(t)) = 1$.

3 | A DIRECT DATA-DRIVEN REFORMULATION

The optimization problem in (12) explicitly depends on the reference signal r (see the constraints in equations (12b) and (12e)) and on the (unknown) dynamics of $\mathcal{G}_s$ via the constraints in (12c)-(12d). By keeping the dependence on r, Problem (12) would need to be solved for different reference trajectories, thus making the approach not feasible in practice in many applications. At the same time, the constraints in equations (12c) and (12d) would require the identification of a model for $\mathcal{G}_s$ prior to the controller design phase. Therefore, before solving the design problem, we annihilate these dependences as described in this section, to reformulate Problem 1 in a purely direct data-driven fashion.

3.1 | Removing the dependence on the reference signal *r*

Through the definitions of the mismatch error in (11) and the left-inverse $M^{\dagger}(s_M(t))$, a *fictitious reference* signal can be obtained as

$$r(t) = M^{\dagger}(s_M(t))y_o(t) - M^{\dagger}(s_M(t))\varepsilon(t). \tag{13}$$

The new reference signal in (13) is a combination of the output $M^{\dagger}(s_M(t))y_o(t)$, returned by the left inverse when fed with the system output $y_o(t)$, and a term compensating for the mismatch error between the desired closed-loop behavior and the attained one. This alternative definition of the reference allows us to express the tracking error as

$$e(t) = r(t) - y_o(t) = M^{\dagger}(s_M(t))y_o(t) - M^{\dagger}(s_M(t))\varepsilon(t) - y_o(t), \tag{14}$$

which is independent of the actual reference signal $r(t)$. Nonetheless, the tracking error depends on both the noiseless output of $\mathcal{G}_s$ and the mismatch error $\varepsilon(t)$.

By exploiting equations (13)-(14), the optimization problem in (12) can be recast into the following reference-free problem:

$$\underset{\varepsilon,\Theta,\{\mathcal{X}_i^c\}_{i=1}^{K_c},\mathcal{S}_c}{\text{minimize}} \|\varepsilon\|_{\ell_2}^2 \tag{15a}$$

$$s.t.\;\; A(s_o(t),q^{-1})y_o(t) = B(s_o(t),q^{-1})u(t) + f(s_o(t)), \;\; \forall t \in \mathcal{I}_1^T \tag{15b}$$

$$s_o(t) = k \iff X_o(t) \in \mathcal{X}_k, \quad k \in \{1,\dots,K\}, \;\; \forall t \in \mathcal{I}_1^T, \tag{15c}$$

$$A_c(s_c(t),q^{-1})u(t) = B_c(s_c(t),q^{-1})\left[M^{\dagger}(s_M(t))y_o(t) - M^{\dagger}(s_M(t))\varepsilon(t) - y_o(t)\right] + f_c(s_c(t)), \quad \forall t \in \mathcal{I}_1^T \tag{15d}$$

$$s_c(t) = i \iff \chi(t) \in \mathcal{X}_i^c, \quad i \in \{1,\dots,K_c\}, \quad \forall t \in \mathcal{I}_1^T, \tag{15e}$$

where the constraint on the mismatch error (see (12b)) is removed since the definition of $\varepsilon(t)$ is used to construct the fictitious reference and the tracking error.

3.2 | Removing the dependence on the dynamics of $\mathcal{G}_s$

By looking at the constraints in (15b)-(15c), it is clear that the problem in (15) still depends on the *unknown* dynamics of the system $\mathcal{G}_s$. The current problem formulation thus requires the identification of the system prior to the controller design phase. To drop the dependence on the model of $\mathcal{G}_s$, we leverage on the available data set $\mathcal{D}_T = \{u(t), y(t)\}_{t=1}^T$. Specifically, instead of explicitly considering the dynamics of the system, the available data are used to replace $y_o(t)$ in (15b)-(15c), so that the resulting problem is given by

$$\underset{\varepsilon,\Theta,\{\mathcal{X}_i^c\}_{i=1}^{K_c},\mathcal{S}_c}{\text{minimize}} \|\varepsilon\|_{\ell_2}^2 \tag{16a}$$

$$\text{s.t. } A_c(s_c(t),q^{-1})u(t) = B_c(s_c(t),q^{-1})\left[M^\dagger(s_M(t))y(t) - M^\dagger(s_M(t))\varepsilon(t) - y(t)\right] + f_c(s_c(t)), \tag{16b}$$

$$s_c(t) = i \iff \chi(t) \in \mathcal{X}_i^c, \quad i \in \{1, \dots, K_c\}, \quad \{u(t), y(t)\} \in \mathcal{D}_T. \tag{16c}$$

The problem to be solved now depends only on the data and the parameters of the controller C_s to be designed. Nonetheless, the problem in (16) is biconvex, due to the products between the optimization variables in the constraint (16b).

Let $\xi(t) \in \mathbb{R}$ be defined as

$$\xi(t) = (M^\dagger(s_M(t)) - 1)y(t), \tag{17}$$

which can be computed relying on available data and the model to be matched. By properly rearranging terms in (16b), it holds that

$$B_c(s_c(t),q^{-1})M^\dagger(s_M(t))\varepsilon(t) = B_c(s_c(t),q^{-1})(M^\dagger(s_M(t)) - 1)y(t) - A_c(s_c(t),q^{-1})u(t) + f_c(s_c(t)), \tag{18}$$

and, by introducing the regressor $X_c(\xi(t)) \in \mathbb{R}^{n_\theta}$, ie,

$$X_c(\xi(t)) = \left[-u(t-1), \ \dots \ -u\left(t-n_a^c\right), \ \xi(t), \ \xi(t-1), \ \dots \ \xi\left(t-n_b^c\right), \ 1\right]', \tag{19}$$

it is straightforward to prove that equation (18) is equivalent to

$$B_c(s_c(t),q^{-1})M^\dagger(s_M(t))\varepsilon(t) = X_c(\xi(t))'\theta_{s_c(t)} - u(t). \tag{20}$$

The design problem is thus recast as the minimization of the parameter-dependent residuals $B_c(s_c(t),q^{-1})M^\dagger(s_M(t))\varepsilon(t)$, ie,

$$\underset{\Theta,\{\mathcal{X}_i^c\}_{i=1}^{K_c},\mathcal{S}_c}{\text{minimize}} \sum_{t=1}^{T}\left(X_c(\xi(t))'\theta_{s_c(t)} - u(t)\right)^2 + \lambda\|\Theta\|_2^2 \tag{21a}$$

$$\text{s.t.} \quad s_c(t) = i \iff \chi(t) \in \mathcal{X}_i^c, \quad i \in \{1, \dots, K_c\}, \quad \{u(t), y(t)\} \in \mathcal{D}_T. \tag{21b}$$

Since our aim is to minimize the mismatch error, it is worth remarking that this problem has to be solved avoiding the trivial solution $B_c(k,q^{-1}) = 0$, for all $k = 1, \dots, K_c$. This goal is accomplished by resorting to an *instrumental variable scheme* when the parameters Θ are estimated, as detailed in Section 4.1.

The $\mathcal{L}_2$ penalty term $\lambda\|\Theta\|_2^2$ is introduced to improve the statistical performance of the controller tuning strategy,[55] with the regularization parameter $\lambda \in \mathbb{R}^+$ chosen to trade-off variance and bias of the estimated parameters.[56] Notice that the higher λ is, the more the local parameters are *shrunk* towards each other and towards zero.[53] Therefore, improper choices of this parameter might lead to a poor matching of the desired closed-loop behavior which, in turn, might cause a deterioration of the tracking performance.

3.3 | Explicit estimation of the polyhedral partition of space $\mathcal{X}^c$

Constraint (21b) in problem (21) depends on the polyhedral partition of the space $\mathcal{X}^c$ of vectors $\chi(t)$ in (10f). However, the partition is *unknown* and, thus, it has to be retrieved from data. Inspired by,[23] a possible approach to handle the estimation of the polyhedral partition is to search for a PWA separator of the domain $\mathcal{X}^c$.

The problem of learning the polyhedral partition from data reduces to find a function $\phi : \mathbb{R}^{n_\chi} \to \mathbb{R}$ defined as

$$\phi(\chi) = \max_{i=1,\dots,K_c} \phi_i(\chi), \tag{22}$$

where $\phi_i(\chi) : \mathbb{R}^{n_\chi} \to \mathbb{R}$, for $i = 1, \dots, K_c$, are affine functions in χ, ie,

$$\phi_i(\chi) = (\omega_i)'\chi - \gamma_i, \tag{23}$$

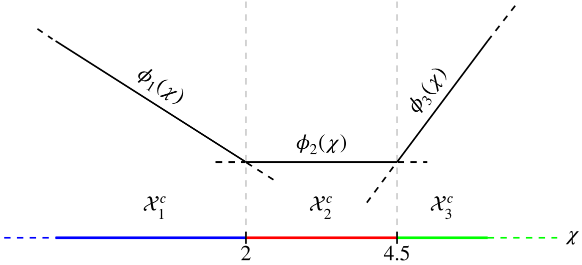

FIGURE 3 Example of three linearly separable clusters and the associated piecewise affine separator [Colour figure can be viewed at wileyonlinelibrary.com]

and the parameter vectors $\omega_i \in \mathbb{R}^{n_\chi}$ and the scalars γ_i have to be estimated from data, for $i = 1, \dots, K_c$. According to the definition of the separator in (22), the i-th polyhedral region $\mathcal{X}_i^c$ is given by the data points satisfying $\phi(\chi) = \phi_i(\chi)$, namely, the points χ such that

$$(\omega_i)'\chi - \gamma_i > (\omega_j)'\chi - \gamma_j \Rightarrow (\omega_j - \omega_i)'\chi - (\gamma_j - \gamma_i) + 1 \leq 0. \tag{24}$$

Example 2. Consider the clusters $\{\mathcal{X}_i^c\}_i^3$ shown in Figure 3, which are defined over the monodimensional space $\mathcal{X}^c$. These clusters are pairwise linearly separable, with the PWA separator $\phi(\chi)$ dividing the clusters also shown in Figure 3. By looking at the separator, intuitively each data point is associated to the corresponding cluster by searching for the maximum affine function among $\{\phi_i\}_{i=1}^3$.

We modify problem (21) to penalize the misclassification of points in the $\mathcal{X}^c$-domain, according to the condition in (24) and for the mode sequence $\mathcal{S}_c$. Specifically, the squared 2-norm of the violation of (24) is introduced in the objective function in (21), and the design problem is recast as follows:

$$\underset{\Theta,\{\omega_i,\gamma_i\}_{i=1}^{K_c},\mathcal{S}_c}{\text{minimize}} \quad \sum_{t=1}^{T}\left(X_c(\xi(t))'\theta_{s_c(t)} - u(t)\right)^2 + \lambda\|\Theta\|_2^2 + \sum_{t=1}^{T}\sum_{\substack{j=1\\ j\neq s_c(t)}}^{K_c}\left\|\left(\left[\,\chi(t)',\ -1\,\right]\begin{bmatrix}\omega_j - \omega_{s_c(t)}\\ \gamma_j - \gamma_{s_c(t)}\end{bmatrix} + 1\right)_+\right\|_2^2 + \lambda_p\sum_{i=1}^{K_c}\left(\|\omega_i\|_2^2 + \gamma_i^2\right). \tag{25}$$

The additional regularization term $\lambda_p\sum_{i=1}^{K_c}(\|\omega_i\|_2^2 + \gamma_i^2)$ is introduced to better condition the multicategory discrimination problem. The finite regularization parameter $\lambda_p \in \mathbb{R}^+$ has to be tuned by the user, considering that λ_p might influence the generalization capabilities of the PWA separator. Indeed, different choices of λ_p might cause translations and/or rotations of the PWA separator. In turn, this leads to changes in the partition induced on $\mathcal{X}^c$, that ultimately results in different switching policies. Therefore, an improper tuning of this parameter might cause a poor matching of $\mathcal{M}_s$.

4 | THE DESIGN METHOD

The data-driven design of the PWA controller in (10a) is carried out by solving the convex unconstrained optimization problem in (25) with respect to the overall parameters Θ of the local controllers, the parameters $\{\omega_i, \gamma_i\}_{i=1}^{K_c}$ of the PWA separator, and the active mode sequence $\mathcal{S}_c$. Inspired by,[49] problem (25) is solved via a coordinate descent method, which alternates between the minimization of the cost with respect to the parameters Θ and $\{\omega_i, \gamma_i\}_{i=1}^{K_c}$, for a fixed sequence $\mathcal{S}_c^{k-1}$, and the optimization with respect to the mode sequence $\mathcal{S}_c$, for the updated parameters Θ^k and $\{\omega_i^k, \gamma_i^k\}_{i=1}^{K_c}$.

The approach, summarized in Algorithm 1, is developed by exploiting the separability of the cost with respect to Θ and the parameters characterizing the PWA separator. Indeed, for a fixed mode sequence $\mathcal{S}_c^{k-1}$, it is straightforward to see that the cost to be optimized, given by

$$\begin{aligned}\mathcal{J}\left(\mathcal{D}_T, \Theta, \{\omega_i,\gamma_i\}_{i=1}^{K_c}, \mathcal{S}_c^{k-1}\right) =& \sum_{t=1}^{T}\left(X_c(\xi(t))'\theta_{s_c^{k-1}(t)} - u(t)\right)^2 + \lambda\|\Theta\|_2^2 + \\ &+ \sum_{t=1}^{T}\sum_{\substack{j=1\\ j\neq s_c^{k-1}(t)}}^{K_c}\left\|\left(\left[\,\chi(t)',\ -1\,\right]\begin{bmatrix}\omega_j - \omega_{s_c^{k-1}(t)}\\ \gamma_j - \gamma_{s_c^{k-1}(t)}\end{bmatrix} + 1\right)_+\right\|_2^2 + \lambda_p\sum_{i=1}^{K_c}\left(\|\omega_i\|_2^2 + \gamma_i^2\right)\end{aligned} \tag{26a}$$

can be split into the sum of the following terms:

$$\mathcal{J}_1\left(\mathcal{D}_T, \Theta, \mathcal{S}_c^{k-1}\right) = \sum_{t=1}^{T}\left(X_c(\xi(t))'\theta_{s_c^{k-1}(t)} - u(t)\right)^2 + \lambda\|\Theta\|_2^2, \tag{26b}$$

$$\mathcal{J}_2\left(\mathcal{D}_T, \{\omega_i, \gamma_i\}_{i=1}^{K_c}, \mathcal{S}_c^{k-1}\right) = \sum_{t=1}^{T}\sum_{\substack{j=1 \\ j\neq s_c^{k-1}(t)}}^{K_c}\left\|\left(\left[\,\chi(t)',\ -1\,\right]\begin{bmatrix}\omega_j - \omega_{s_c^{k-1}(t)} \\ \gamma_j - \gamma_{s_c^{k-1}(t)}\end{bmatrix} + 1\right)_+\right\|_2^2 + \lambda_p\sum_{i=1}^{K_c}\left(\|\omega_i\|_2^2 + \gamma_i^2\right), \tag{26c}$$

allowing us to separately estimate Θ and $\{\omega_i, \gamma_i\}_{i=1}^{K_c}$. The update of the active mode sequence at Step 1.3 is performed accounting for both the continuous dynamics of the local controllers and the position of the samples with respect to the estimated partition. It is worth remarking that the estimate of $\mathcal{S}_c$ allows us to cluster the available data and, thus, to solve the problems at Steps 1.1 and 1.2 in a supervised fashion. Algorithm 1 is run offline until either the cost $\mathcal{J}$ does not decrease *significantly* (according to a tunable threshold $\epsilon_J > 0$) within two consecutive iterations (27) or the maximum number of iterations k_{max} is attained. This last stopping criterion is introduced to guarantee the termination of Algorithm 1 in finite time, but it does not affect the closed-loop performance attained with the learned controller $\mathcal{C}_s$. Indeed, if the problem is well-posed, Algorithm 1 usually converges to a solution within few steps, thus satisfying the first stopping criterion.*

Algorithm 1 [Coordinate descent approach for PWA controller design]

Input: Dataset set $\mathcal{D}_T = \{u(t), y(t)\}_{t=1}^{T}$; initial mode sequence $\mathcal{S}_c^0$; tolerance ϵ_J; maximum number of iterations k_{max}.

```
1. for $k = 1, \ldots$ do
    1.1 $\Theta^k \leftarrow \arg\min_{\Theta} \mathcal{J}_1(\mathcal{D}_T, \Theta, \mathcal{S}_c^{k-1})$;    (local model fitting)
    1.2 $\{\omega_i^k, \gamma_i^k\}_{i=1}^{K_c} \leftarrow \arg\min_{\{\omega_i,\gamma_i\}_{i=1}^{K_c}} \mathcal{J}_2(\mathcal{D}_T, \{\omega_i, \gamma_i\}_{i=1}^{K_c}, \mathcal{S}_c^{k-1})$;    (computation of the piece-wise affine separator)
    1.3 $\mathcal{S}_c^k \leftarrow \arg\min_{\mathcal{S}_c} \mathcal{J}(\mathcal{D}_T, \Theta^k, \{\omega_i^k, \gamma_i^k\}_{i=1}^{K_c}, \mathcal{S}_c)$;    (mode sequence update)
2. until the maximum number of iterations $k_{max}$ is attained or
    $|\mathcal{J}(\mathcal{D}_T, \Theta^k, \{\omega_i^k, \gamma_i^k\}_{i=1}^{K_c}, \mathcal{S}_c^k) - \mathcal{J}(\mathcal{D}_T, \Theta^{k-1}, \{\omega_i^{k-1}, \gamma_i^{k-1}\}_{i=1}^{K_c}, \mathcal{S}_c^{k-1})| < \epsilon_J$;    (27)
```

Output: Estimated parameters $\Theta^\star = \Theta^k$ of the local controllers; estimated parameters $\{\omega_i^\star, \gamma_i^\star\}_{i=1}^{K_c} = \{\omega_i^k, \gamma_i^k\}_{i=1}^{K_c}$ of the piece-wise affine separator; estimated mode sequence $\mathcal{S}_c^\star = \mathcal{S}_c^k$.

Remark 4. Once Algorithm 1 is stopped, the stability of the resulting controller can be checked by resorting to its nonminimal state-space representation.[28] Accordingly, the model-based approach proposed in[57] or the data-driven method presented in[58,59] can be exploited to study the stability of $\mathcal{C}_s^\star$. However, provided that only input/output data are available in closed-loop, to the best of the authors knowledge, no input/output criterion exists to check the stability of the closed-loop system once $\mathcal{C}_s^\star$ has been learned. Despite this limitation, if the problem is well-posed and a suitable *stable* reference model is chosen, the closed-loop system is expected to match the desired behavior.

Once Algorithm 1 is terminated, the resulting controller $\mathcal{C}_s^\star$ can be used to control $\mathcal{G}_s$, with the control action $\tilde{u}(t)$ fed to the system obtained as described in Algorithm 2. In particular, the active local controller is chosen according to the estimated polyhedral partition of space $\mathcal{X}^c$ (Step 2) and, then, the input is generated by using the estimated parameters for the chosen local controller (Step 4). Note that the reference signal $\tilde{r}$ feeding the closed-loop system is not the fictitious reference used for training, but it is the one selected by the user according to the control task at hand.

*Only if the problem is ill-posed or the chosen ϵ_J is excessively small, the first stopping criterion might be never satisfied. To avoid infinite loops, it is thus reasonable to set $k_{\text{max}} = 100$, so to always terminate the iterations, while restraining the computational time in case Algorithm 1 does not converge.

Algorithm 2 [PWA control signal selection]

Input: Measured closed-loop output $\tilde{y}(t)$; past input/closed-loop-output pairs $\{\tilde{u}(\tau), \tilde{y}(\tau)\}_{\tau=1}^{t}$; reference signal $\tilde{r}(t)$; parameters $\Theta^{\star}$ of the local controllers; parameters $\{\omega_i^{\star}, \gamma_i^{\star}\}_{i=1}^{K_c}$ of the piece-wise affine separator.

```
1. construct the vector $\tilde{\chi}(t) \in \mathbb{R}^{\chi}$ as in (10f);
2. find the local controller to be used as
       $\tilde{s}_c(t) \leftarrow \arg\max_{i=1,\dots,K^c} (\omega_i^{\star})' \tilde{\chi}(t) - \gamma_i^{\star};$   (28)
3. construct the regressor $\tilde{X}_c(t)$ as in (19), with $\xi(\tau)$ replaced by the tracking error $\tilde{e}(\tau) = \tilde{r}(t) - \tilde{y}(\tau)$, $\tau = t - n_b^c, \dots, t$;
4. generate the input $\tilde{u}(t)$ as
       $\tilde{u}(t) \leftarrow \tilde{X}_c(t)' \theta^{\star}_{\tilde{s}_c(t)}.$   (29)
```

Output: Controller output $\tilde{u}(t)$.

4.1 | Step 1.1: estimation of the parameters Θ

For a fixed mode sequence $\mathcal{S}_c^{k-1}$, the estimation of the parameters Θ reduces to the solution of the optimization problem

$$\min_{\Theta} \mathcal{J}_1 \left(\mathcal{D}_T, \Theta, \mathcal{S}_c^{k-1} \right), \tag{30}$$

with the objective function defined as in (26b).

Assume that the sequence $\mathcal{S}_{o,c}$ and the partition $\{\mathcal{X}_{o,i}^c\}_{i=1}^{K_c}$ satisfying Assumption 1 are known and $\mathcal{S}_c^{k-1} = \mathcal{S}_{o,c}$. If the output measurements are not corrupted by noise (ie, $v(t) = 0$ in (9)), problem (30) is equivalent to the one in Equation (15). In this case $X_c(\xi(t))$ in (26b) is simply replaced with $X_c(\xi_o(t))$, with the noiseless regressor given by

$$X_c(\xi_o(t)) = \left[-u(t-1), \ \cdots \ -u\left(t - n_a^c\right), \ \xi_o(t), \ \xi_o(t-1), \ \cdots \ \xi_o\left(t - n_b^c\right), \ 1 \right]', \tag{31}$$

and

$$\xi_o(t) = (M^{\dagger}(s_M(t)) - 1) y_o(t). \tag{32}$$

However, in realistic scenarios where only noisy data are available, the solution of problem (30) is biased even if $\{\mathcal{X}_{o,i}^c\}_{i=1}^{K_c}$ and $\mathcal{S}_{o,c}$ are known, as shown in the next section. This discrepancy between the solution of (12) and (25) is handled via a proper stochastic treatment of noise, by recasting the cost $\mathcal{J}_1(\mathcal{D}_T, \Theta, \mathcal{S}_c^{k-1})$ via an *IV* scheme. We thus introduce an *instrument* $\zeta(t) \in \mathbb{R}^{n_\theta}$, with the same dimension of the regressor $X_c(\xi(t))$, which is selected so that it is not correlated with the noise dependent term $\xi - \xi_o = (M^{\dagger}(s_M(t)) - 1)v(t)$.

Remark 5. The instrument can be built by constructing a new set of regressors through a new experiment carried out on $\mathcal{G}_s$ with the same input used to generate $\mathcal{D}_T$. Since the output measurements collected during the new experiment are affected by a different realization of the noise, this should guarantee that $\zeta(t)$ is correlated with the system dynamics. At the same time, the instrument should be uncorrelated with $(M^{\dagger}(s_M(t)) - 1)v(t)$.

By introducing $\zeta(t)$, problem (30) is reformulated as

$$\min_{\Theta} \left\| \sum_{t=1}^{T} \zeta(t) \left(X_c(\xi(t))' \theta_{s_c^{k-1}(t)} - u(t) \right) \right\|_2^2 + \lambda \|\Theta\|_2^2. \tag{33a}$$

Note that, differently from $\mathcal{J}_1(\mathcal{D}_T, \Theta, \mathcal{S}_c^{k-1})$ in (26b), the objective function in (33) is not separable with respect to time. Therefore, the parameters θ_i, with $i = 1, \dots, K_c$, cannot be updated by solving K_c separate optimization problems.

Although this impacts to a certain extent on the computational time required to design the controller, this is not critical since we are designing the controller offline.

Let $\mathbb{I}_{[s_c^{k-1}(t)]} \in \{0,1\}^{n_\theta \times n_\Theta}$ be defined as

$$\mathbb{I}_{[s_c^{k-1}(t)]} = \left[I_{n_\Theta} \mathbb{1}_{[s_c^{k-1}(t)=1]}, \ \ldots, \ I_{n_\Theta} \mathbb{1}_{[s_c^{k-1}(t)=K_c]} \right], \tag{34}$$

with $\mathbb{1}_{[s_c^{k-1}(t)=i]} = 1$ if and only if $s_c^{k-1}(t) = i$. Problem (33) can be equivalently reformulated as

$$\min_{\Theta} \left\| \sum_{t=1}^{T} \zeta(t) \left(X_c(\xi(t))' \mathbb{I}_{[s_c^{k-1}(t)]} \Theta - u(t) \right) \right\|_2^2 + \lambda \|\Theta\|_2^2, \tag{35a}$$

which explicitly depends on the unknown parameter matrix Θ. We can then construct $Z \in \mathbb{R}^{T \times n_\theta}$, $U \in \mathbb{R}^T$, and $R(\xi) \in \mathbb{R}^{T \times n_\Theta}$, which are obtained by stacking the elements of $\{\zeta(t)\}_{t=1}^T$, $\{u(t)\}_{t=1}^T$ and $\{X_c(\xi(t))' \mathbb{I}_{[s_c^{k-1}(t)]}\}_{t=1}^T$ as

$$Z = \begin{bmatrix} \zeta(1)' \\ \vdots \\ \zeta(T)' \end{bmatrix}, \qquad U = \begin{bmatrix} u(1) \\ \vdots \\ u(T) \end{bmatrix}, \qquad R(\xi) = \begin{bmatrix} X_c(\xi(1))' \mathbb{I}_{[s_c^{k-1}(1)]} \\ \vdots \\ X_c(\xi(T))' \mathbb{I}_{[s_c^{k-1}(T)]} \end{bmatrix}. \tag{36}$$

Problem (35) can thus be recast in matrix form as

$$\min_{\Theta} \|Z'(R(\xi)\Theta - U)\|_2^2 + \lambda \|\Theta\|_2^2, \tag{37}$$

whose closed-form solution is given by

$$\Theta^k = \left((Z'R(\xi))' Z'R(\xi) + \lambda I_{n_\Theta} \right)^{-1} (Z'R(\xi))' Z'U. \tag{38a}$$

Beside the impact of the $\mathcal{L}_2$ penalty introduced in (21) on the statistical properties of the designed controller, it is worth remarking that this regularization term causes the addition of $\lambda \geq 0$ to all the diagonal terms of $(Z'R(\xi))'Z'R(\xi)$, so that the matrix to be inverted is always nonsingular. Therefore, problem (37) is computationally tractable even if $(Z'R(\xi))'Z'R(\xi)$ is not full rank.

Once Θ^k is computed, the estimated parameters for the ith local controller can simply be retrieved as

$$\theta_i^k = \mathbb{I}_{[i]} \Theta^k. \tag{38b}$$

Notice that the computation of Θ^k in (38a) requires the inversion of an n_Θ-dimensional matrix and, thus, the computational complexity of the method increases with the number of parameters and the number of possible local controllers. Strategies for complexity reduction will be the subject of future investigations.

4.1.1 | On the consistency of the estimated parameters

Let us assume that the following assumptions are verified.

Assumption 5. Let S_c^{k-1} used to solve problem (37) be different from the sequence $\mathcal{S}_{o,c}$ satisfying Assumption 1. Denote with $M = \{t \, : \, s_c^{k-1}(t) \neq s_{o,c}(t)\}$ the set of *misclassified points* and with $\#m \in \mathbb{N}$ the cardinality of M. It holds that $\#m \ll T$, with T being the length of the data set $\mathcal{D}_T$.

Assumption 6. The IV is properly selected, namely it is such that

$$\frac{1}{T} \sum_{t=1}^{T} \zeta(t) X_c(\xi(t)) \mathbb{I}_{[s_c^{k-1}(t)]} \underset{T\to\infty}{\rightarrow} \mathbb{E}\left[\zeta(t) X_c(\xi(t)) \mathbb{I}_{[s_c^{k-1}(t)]} \right] \neq 0, \tag{39a}$$

$$\frac{1}{T} \sum_{t=1}^{T} \zeta(t) X_c(\xi(t)) \mathbb{I}_{[s_c^{k-1}(t)]} \underset{T\to\infty}{\rightarrow} \mathbb{E}\left[\zeta(t) X_c(\xi(t)) \mathbb{I}_{[s_c^{k-1}(t)]} \right] < \infty, \tag{39b}$$

$$\frac{1}{T}\sum_{t=1}^{T}\zeta(t)X_c(\xi(t))\mathbb{I}_{[s_{o,c}(t)]}\underset{T\to\infty}{\to}\mathbb{E}[\zeta(t)X_c(\xi(t))\mathbb{I}_{[s_{o,c}(t)]}]<\infty, \tag{39c}$$

$$\frac{1}{T}\sum_{t=1}^{T}\zeta(t)(\xi(t)-\xi_o(t))\mathbb{I}_{[s_c^{k-1}(t)]}\underset{T\to\infty}{\to}\mathbb{E}\left[\zeta(t)(\xi(t)-\xi_o(t))\mathbb{I}_{[s_c^{k-1}(t)]}\right]=0. \tag{39d}$$

Conditions (39a)-(39d) entail a proper choice of the instrument and are commonly exploited to prove consistency.[60] Under these conditions, we can prove the following consistency result on the parameters estimated as in (38a).

Theorem 1. *If Assumptions 1-6 hold, then*

1. *the estimate Θ^k obtained by solving problem (25) is biased, even if the partition $\{\mathcal{X}_i^c\}_{i=1}^{K_c}$ and the mode sequence $\mathcal{S}_{o,c}$ satisfying Assumptions 1-3 are known;*
2. *the estimate Θ^k computed as in (38a) asymptotically converges to the parameter matrix Θ_o satisfying Assumptions 1-3, ie,*

$$\lim_{T\to\infty}\Theta^k=\Theta_o. \tag{40}$$

Proof. See the Appendix. □

4.2 | Step 1.2: construction of the polyhedral partition of $\mathcal{X}^c$

Since the mode sequence is fixed to $\mathcal{S}_c^{k-1}$, the data can be split into K_c distinct set of points $C_i=\{\chi(t), t\in\mathcal{I}_T\ :\ s_c^{k-1}(t)=i\}$, for $i=1,\dots,K_c$. By relying on these clusters, the PWA separator defined in (22) is estimated via the optimization of $\mathcal{J}_2(\mathcal{D}_T,\{\omega_i,\gamma_i\}_{i=1}^{K_c})$ defined in (26c). Specifically, the cost function in equation (26c) is equivalent to

$$\mathcal{J}_2\left(\mathcal{D}_T,\{\omega_i,\gamma_i\}_{i=1}^{K_c},\mathcal{S}_c^{k-1}\right)=\sum_{i=1}^{K_c}\ \sum_{t:\ s_c^{k-1}(t)\in C_i}\ \sum_{\substack{j=1\\j\neq i}}^{K_c}\left\|\left(\left[\ \chi(t)',\ -1\ \right]\begin{bmatrix}\omega_j-\omega_i\\\gamma_j-\gamma_i\end{bmatrix}+1\right)_+\right\|_2^2+\lambda_p\sum_{i=1}^{K_c}\left(\|\omega_i\|_2^2+\gamma_i^2\right), \tag{41}$$

where we explicitly introduce the dependence of the cost on the number of misclassified points for each cluster.

Let c_i denote the cardinality of the i-th cluster, namely, the number of points associated to C_i, for $i=1,\dots,K_c$. As the problem is solved offline, we construct the vector X_i stacking the elements of the i-th cluster, so that the cost to be optimized is equal to

$$\mathcal{J}_2\left(\mathcal{D}_T,\{\omega_i,\gamma_i\}_{i=1}^{K_c},\mathcal{S}_c^{k-1}\right)=\sum_{i=1}^{K_c}\sum_{\substack{j=1\\j\neq i}}^{K_c}\frac{1}{c_i}\left\|\left(\left[\ X_i',\ -\mathbf{1}_{c_i}\ \right]\begin{bmatrix}\omega_j-\omega_i\\\gamma_j-\gamma_i\end{bmatrix}+\mathbf{1}_{c_i}\right)_+\right\|_2^2+\lambda_p\sum_{i=1}^{K_c}\left(\|\omega_i\|_2^2+\gamma_i^2\right), \tag{42}$$

where the normalization with respect to c_i (the number of points associated to each cluster) is introduced for the problem to be better conditioned. This cost function allows us to find the PWA separator via the regularized piecewise smooth Newton method with Armijo's line search proposed in,[23] which significantly reduces the computational time required to solve the discrimination problem with respect to other multicategory classification methods, even when large data sets are used.

4.3 | Step 1.3: update of the active mode sequence $\mathcal{S}_c$

Once the parameters Θ^k and $\{\omega_i^k,\gamma_i^k\}_{i=1}^{K_c}$ have been computed, the active mode sequence $\mathcal{S}_c$ can be updated by minimizing the cost $\mathcal{J}(\mathcal{D}_T,\Theta^k,\{\omega_i^k,\gamma_i^k\}_{i=1}^{K_c},\mathcal{S}_c)$ in (26a). The sequence $\mathcal{S}_c$ is thus estimated by solving the optimization problem

$$\min_{\mathcal{S}_c}\sum_{t=1}^{T}\left[\left(X_c(\xi(t))'\theta_{s_c(t)}^k-u(t)\right)^2+\sum_{\substack{j=1\\j\neq s_c(t)}}^{K_c}\left\|\left(\left[\ \chi(t)',\ -1\ \right]\begin{bmatrix}\omega_j^k-\omega_{s_c(t)}^k\\\gamma_j^k-\gamma_{s_c(t)}^k\end{bmatrix}+1\right)_+\right\|_2^2\right], \tag{43}$$

where the regularization terms have been removed as they are independent of the mode sequence. Since the cost function in (43) is separable over the time, the estimation of the mode sequence can be solved to achieve global optimality by *dynamic programming*[61] with complexity $O(TK_c^2)$. This is attained as follows.

Let $\mathcal{J}_t(u(t), y(t), \theta_i^{s_c(t)}, \omega_i^{s_c(t)}, \gamma_i^{s_c(t)}) \in \mathbb{R}$ be the t-th term of the objective function in (43), ie,

$$\mathcal{J}_t\left(u(t), y(t), \theta^k_{s_c(t)}, \omega^k_{s_c(t)}, \gamma^k_{s_c(t)}\right) = \left(X_c(\xi(t))'\theta^k_{s_c(t)} - u(t)\right)^2 + \sum_{\substack{j=1 \\ j\neq i}}^{K_c} \left\| \left([\ \chi(t)',\ -1\ ] \begin{bmatrix} \omega^k_j - \omega^k_{s_c(t)} \\ \gamma^k_j - \gamma^k_{s_c(t)} \end{bmatrix} + 1 \right)_+ \right\|_2^2 . \tag{44}$$

First, the *terminal cost* $J(i, T) \in \mathbb{R}$ is computed for all $i \in \{1, \dots, K_c\}$ as

$$J(i, T) = \mathcal{J}_T\left(u(T), y(T), \theta^k_i, \omega^k_i, \gamma^k_i\right). \tag{45}$$

Then, the *cost* $J(i, t) \in \mathbb{R}$ and the mode $S(i, t+1) \in \mathbb{R}$ are obtained as

$$J(i, t) = \mathcal{J}_t\left(u(t), y(t), \theta^k_i, \omega^k_i, \gamma^k_i\right) + \min_j J(j, t+1) \tag{46}$$

backward in time, for $t = T-1, \dots, 1$. The mode sequence is then reconstructed forward in time as

$$s_c^k(t) = \arg\min_{i=1,\dots,K_c} J(i, t), \quad t = 1, \dots, T. \tag{47}$$

Note that the optimal active mode at time t is selected as the one that trades off input matching ($u(t) \approx X_c(\xi(t))'\theta_{s_c^k(t)}$) and the assignment of $\chi(t)$ to the polyhedral region such that

$$\left(\omega_j - \omega_{s_c^k(t)}\right)'\chi(t) - \left(\gamma_j - \gamma_{s_c^k(t)}\right) + 1 \leq 0, \quad \forall j \neq s_c^k(t). \tag{48}$$

Therefore, this *clustering* policy allows us to account for both the local dynamics of the controller and the hypothesis made on the discrete dynamics of $\mathcal{C}_s$.

5 | ON THE COMPUTATION OF THE FICTITIOUS REFERENCE SIGNAL

As outlined in Section 3.1, the dependence of the model-matching design problem (12) on the reference signal r is annihilated by constructing a *fictitious reference* signal from data. Such a signal should not be confused with the *virtual reference* signal computed in the VRFT algorithm.[31] In our case, the fictitious signal embeds the mismatch ϵ to be minimized, while in VRFT the virtual reference represents the "ideal" signal that would feed the closed-loop if the latter coincides with the desired model. Further comments about the differences between our optimization-based approach and state-of-the-art data-driven control can be found in.[48]

Note that the derivation of r requires the computation of the left-inverse of the PWA reference model $\mathcal{M}$ in (7), with the left-inverse defined as in Definition 2. Since the discrete state $s_M(t) \in \{1, \dots, K_M\}$ of $M(s_M(t))$ can be measured for all $t \in \mathbb{N}$, the problem of computing the left-inverse of the reference mapping $M(s_M(t))$ in (7) reduces to the determination of the left-inverse of an LPV map with *integer* scheduling variable. This holds as the polyhedral partition driving the discrete dynamics depends on the state of $M(s_M(t))$ only (see (7c)), which is invariant to the inversion, so that the discrete state is the same for the reference model $\mathcal{M}_s$ and the left-inverse $M^\dagger(s_M(t))$. Therefore, by relying on existing results on the computation of the left-inverse for LPV mappings,[48] the left-inverse of the PWA map $M(s_M(t))$ can be found according to the following proposition.

Proposition 1. *Consider the state-space reference model in (7) and assume that there exists a finite scalar $D_M(i)^{-1}$ such that $D_M(i)^{-1}D_M(i) = 1$, $\forall i \in \{1, \dots, K_M\}$. Then, the left-inverse of $M(s_M(t))$ can be computed as*

$$x_M(t+1) = A_{M^\dagger}(s_M(t))x_M(t) + B_{M^\dagger}(s_M(t))y_d(t), \tag{49a}$$

$$r(t) = C_{M^\dagger}(s_M(t))x_M(t) + D_{M^\dagger}(s_M(t))y_d(t), \tag{49b}$$

with the matrices of the left-inverse given by

$$A_{M^\dagger}(s_M(t)) = A_M(s_M(t)) - B_M(s_M(t))D_M(s_M(t))^{-1}C_M(s_M(t)), \tag{49c}$$

$$B_{M^\dagger}(s_M(t)) = B_M(s_M(t))D_M(s_M(t))^{-1}, \tag{49d}$$

$$C_{M^\dagger}(s_M(t)) = -D_M(s_M(t))^{-1}C_M(s_M(t)), \tag{49e}$$

$$D_{M^\dagger}(s_M(t)) = D_M(s_M(t))^{-1}. \tag{49f}$$

Proposition 1 can be proven as shown in Formentin et al,[48] with the scheduling variable replaced by the discrete state $s_M(t)$.

If at least one of the submodels of $\mathcal{M}$ in (7) is strictly proper, namely there exists $i \in \{1, \dots, K_M\}$ such that $D_M(i) = 0$, then the left-inverse of the i-th submodel cannot be computed by directly applying Proposition 1. A possible approach to overcome this problem is to approximate $D_M(i) = 0$ with $\hat{D}_M(i) = \epsilon_D$, where $0 < \epsilon_D \ll 1$. Alternatively, similarly to what is done in,[48] we can exploit the fact that the controller is designed offline. Indeed, for all $\tau \geq 0$, it is straightforward to prove that the following equality holds:

$$\begin{aligned} y_d(t+\tau) =& C_M(s_M(t+\tau))\prod_{i=0}^{\tau-1}A_M(s_M(t+\tau-i-1))x_M(t)+ \\ &+ C_M(s_M(t+\tau))\sum_{j=0}^{\tau-2}\prod_{i=0}^{j}A_M(s_M(t+\tau-i-1))B_M(s_M(t+\tau-j-2))r(t+\tau-j-2)+ \\ &+ C_M(s_M(t+\tau))B_M(s_M(t+\tau-1))r(t+\tau-1). \end{aligned} \tag{50}$$

Then, by considering the first finite value $\bar{\tau} \in \mathbb{N}$ such that $y_d(t+\bar{\tau})$ depends algebraically on $r(t)$, ie,

$$y_d(t+\bar{\tau}) = C_M(s_M(t+\bar{\tau}))\prod_{i=0}^{\bar{\tau}-1}A_M(s_M(t+\bar{\tau}-i-1))x_M(t) + \prod_{i=0}^{\bar{\tau}-2}A_M(s_M(t+\bar{\tau}-i-1))B_M(s_M(t))r(t), \tag{51}$$

we can define a *shifted* reference model $\mathcal{M}_s'$, described by the difference equations

$$x_M(t+1) = A_M(s_M(t))x_M(t) + B_M(s_M(t))r(t), \tag{52a}$$

$$y_{M'}(t) = C_{M'}(s_M(t))x_M(t) + D_{M'}(s_M(t))r(t), \tag{52b}$$

with $y_{M'}(t) = y_d(t+\bar{\tau})$ and matrices

$$C_{M'}(s_M(t)) = C_M(s_M(t+\bar{\tau}))\prod_{i=0}^{\bar{\tau}-1}A_M(s_M(t+\bar{\tau}-i-1)), \tag{52c}$$

$$D_{M'}(s_M(t)) = \prod_{i=0}^{\bar{\tau}-2}A_M(s_M(t+\bar{\tau}-i-1))B_M(s_M(t)). \tag{52d}$$

The discrete dynamics of the reference model $\mathcal{M}_s'$ is still driven by the polyhedral partition characterizing $\mathcal{M}_s$, as the state of $\mathcal{M}_s'$ and $\mathcal{M}_s$ is equal. Since the matrices $D_{M'}(i)$ of the shifted system in (52) are such that $D_{M'}(i) \neq 0$ for all $i \in \{1, \dots, K_c\}$, the left-inverse of $\mathcal{M}_s'$ can be computed as in Proposition 1. Any signal filtered through $\mathcal{M}_s'$ is simply a delayed version of the output of $\mathcal{M}_s$, thus the latter can be easily reconstructed from the former.

Note that, by using $\mathcal{M}_s'$, the number of samples available for the controller design reduces to $T-\bar{\tau}$, but this is not critical since usually $\bar{\tau} \ll T$.

6 | SIMULATION EXAMPLES

The effectiveness of the proposed data-driven design strategy is evaluated through three numerical examples and a simulation case study of practical interest, already considered in.[48] The first and the second numerical examples concern the control of two PWA systems (with LTI and PWL reference models, respectively), so that the dynamics of the (unknown) plant $\mathcal{G}_s$ is exactly described by Equations (2)-(4). On the other hand, in the other examples, the proposed method is used to design a PWA controller for two different nonlinear systems. In these cases, the behavior of the true system can only be approximated through equations (2)-(4). In all examples, the output measurements are supposed to be corrupted by noise, whose effect is assessed through the *signal-to-noise ratio* (SNR), ie,

$$\text{SNR} = 10\log\frac{\sum_{t=1}^{T} y_o(t)^2}{\sum_{t=1}^{T} v(t)^2}. \tag{53}$$

The instrument ζ is constructed by performing a second open-loop experiment on the system with the same input used to build $\mathcal{D}_T$.

Since, in most applications, the control action feeding the switching/nonlinear systems is aimed at compensating their discontinuous/nonlinear nature, we consider LTI reference models for three out of four case studies. Nonetheless, by selecting a PWL reference model for the second numerical example, the performance of PWA controllers designed directly from data are also assessed in case the ultimate goal is for the closed-loop system to be characterized by different operating regimes. Independently of its characteristics, the reference model $\mathcal{M}_s$ is always chosen to be strictly causal and, thus, it has no direct feed-through. Therefore, its left-inverse is always computed as described in Section 5.

Due to the dependence of the outcome of Algorithm 1 on the initial guess $\mathcal{S}_c^0$ for the mode sequence, Algorithm 1 is carried out for N different initial sequences generated randomly, selecting the solution associated to the least value of the cost $\mathcal{J}$. This should help us avoiding Algorithm 1 to be trapped into a local minimum. For all possible initial mode sequences, Algorithm 1 is run until $k_{\max} = 100$ iterations are carried out or the condition in equation (27) is satisfied with tolerance $\epsilon_J = 10^{-10}$. The polyhedral partition at Step 1.2 of Algorithm 1 is computed by using the piecewise smooth Newton method proposed in,[23] by using the same parameters reported there in.[23]

For each case study, the number of local controllers and the regularization parameters are chosen through cross-validation, by running Algorithm 1 for different combinations of the tuning parameters, with their possible values defined over a preselected grid of values. Cross-validation is performed over a data set $\mathcal{D}_{\bar{T}}$ of length $\bar{T} \ll T$ not used for learning. Since the value of the affine term in the local control law might affect the tracking performance in closed-loop, a different regularization parameter $\lambda_{aff} \neq \lambda$ is considered for the $\mathcal{L}_2$ penalty associated to this term. As an improper tuning might lead to a poor matching of the desired closed-loop behavior, performing closed-loop experiments for each controller might be unsafe. Therefore, the parameters to be used are selected without performing closed-loop experiments, but they are chosen as the ones minimizing the *root-mean-square error* (RMSE) in the reconstruction of the measured input, ie,

$$\left\{K_c^\star, \lambda^\star, \lambda_{aff}^\star, \lambda_p^\star\right\} \leftarrow \arg\min_{Kc,\lambda,\lambda_{aff},\lambda_p} \text{RMSE}_u(K_c, \lambda, \lambda_{aff}, \lambda_p), \tag{54}$$

with RMSE_u defined as follows:

$$\text{RMSE}_u = \sqrt{\frac{1}{\bar{T}}\sum_{t=1}^{\bar{T}}(u(t) - \bar{u}(t; K_c, \lambda, \lambda_{aff}, \lambda_p))^2}, \tag{55}$$

$u(t)$ being the input fed in open-loop to the plant at time t, and $\{\bar{u}(t; K_c, \lambda, \lambda_{aff}, \lambda_p)\}_{t=1}^{\bar{T}}$ being the input obtained by simulating the controller learned for fixed values of K_c, λ, λ_{aff}, and λ_p in open-loop. When performing cross-validation, Algorithm 1 is run for $N = 20$ randomly generated initial guesses for the mode sequence.

Once the tuning parameters are selected and the controller is learned via Algorithm 1, its performance is evaluated with respect to a certain tracking task. In this case, the difference between the desired closed-loop behavior and the one

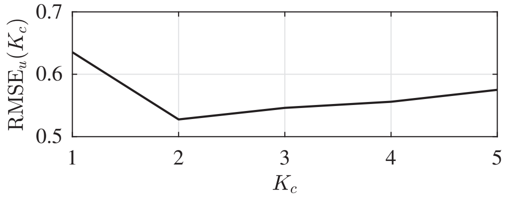

FIGURE 4 Linear time-invariant reference model: $\mathrm{RMSE}_u(K_c)$ vs number of local controllers K_c. RMSE, root-mean-square error

attained with the designed controller is quantitatively assessed through the *RMSE*, ie,

$$\mathrm{RMSE} = \sqrt{\frac{1}{\bar{T}}\sum_{t=1}^{\bar{T}}(\tilde{y}(t) - y_d(t))^2}. \tag{56}$$

All the tests are performed on an i7 2.8-GHz Intel core processor with 16 GB of RAM running MATLAB 2018b.

6.1 | Numerical example: LTI reference model

Consider the PWL system with $K = 2$ modes already introduced in Example 1, ie,

$$x(t+1) = 0.8\begin{bmatrix}\cos\alpha(t) & -\sin\alpha(t)\\ \sin\alpha(t) & \cos\alpha(t)\end{bmatrix}x(t) + \begin{bmatrix}0\\1\end{bmatrix}u(t), \tag{57a}$$

$$y(t) = [\,0\;\;1\,]x(t), \tag{57b}$$

where

$$\alpha(t) = \begin{cases}\frac{\pi}{3} & \text{if}\quad [\,1\;\;0\,]x(t) \geq 0,\\ -\frac{\pi}{3} & \text{if}\quad [\,1\;\;0\,]x(t) < 0.\end{cases} \tag{57c}$$

The desired closed-loop behavior $\mathcal{M}_s$ is chosen as the second-order LTI system described by the difference equation

$$x_M(t+1) = \begin{bmatrix}-1 & 1\\ -0.8 & 1\end{bmatrix}x_M(t) + \begin{bmatrix}0.6\\0.8\end{bmatrix}r(t), \tag{58a}$$

$$y_d(t) = [\,1\;\;0\,]x_M(t). \tag{58b}$$

The PWA controller designed to attain the desired closed-loop behavior is characterized by first-order local controllers with integral action (ie, $n_a^c = n_b^c = 1$ and $a_i(s_c(t)) = 1$ for $i = 1, 2$). The control law C_s is thus given by

$$u(t) = u(t-1) + \theta_{s_c(t),1}(r(t) - y(t)) + \theta_{s_c(t),2}(r(t-1) - y(t-1)) + \theta_{s_c(t),3}. \tag{59}$$

A data set $\mathcal{D}_T$ of length $T = 2750$ is generated by exciting the system in (57) with a sequence of inputs uniformly distributed in the interval $[-1, 1]$. The corresponding outputs are corrupted by a zero-mean white noise $v(t)$ with Gaussian distribution and standard deviation $\sigma_v = 0.35$, which yields SNR=7.1 dB. An additional experiment is performed in similar conditions to generate a data set of length $\bar{T} = 1250$ used for cross-validation. By considering a maximum of five local models, Figure 4 shows the *smallest* RMSE_u among the ones resulting from different choices of the tuning parameters, obtained for different values of K_c. It is clear that the minimum is attained for $K_c^\star = 2$. Unsurprisingly, as the reference model is LTI, this value corresponds to the number of operating conditions of $\mathcal{G}_s$. According to the chosen $K_c^\star$, the other tuning parameters are set to $\lambda^\star = 0.1$, $\lambda_{aff}^\star = 10$ and $\lambda_p^\star = 0.1$.

With the selected tuning parameters, the controller is learned by running Algorithm 1 for $N = 20$ different initial sequences. The partition of the regressor space $\mathcal{X}^c$ characterizing the switching policy of the controller is shown in Figure 5. It is interesting to note that the induced partition clearly differs from the one describing $\mathcal{G}_s$ in Equation (57). Indeed, the discrete dynamics of $\mathcal{G}_s$ is governed by the first component of the plant state, while the controller partition depends on the current/past outputs and past inputs, with the output corresponding to the second component of the state of $\mathcal{G}_s$.

The performance of the controller is initially assessed by simulating the closed-loop system in presence of a piecewise constant reference. In the simulation, the feedback output y is still corrupted by noise, which has the same characteristics

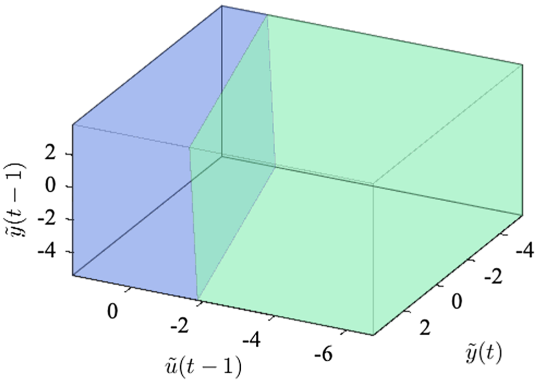

FIGURE 5 Linear time-invariant reference model: estimated polyhedral partition of the domain $\mathcal{X}^c$ [Colour figure can be viewed at wileyonlinelibrary.com]

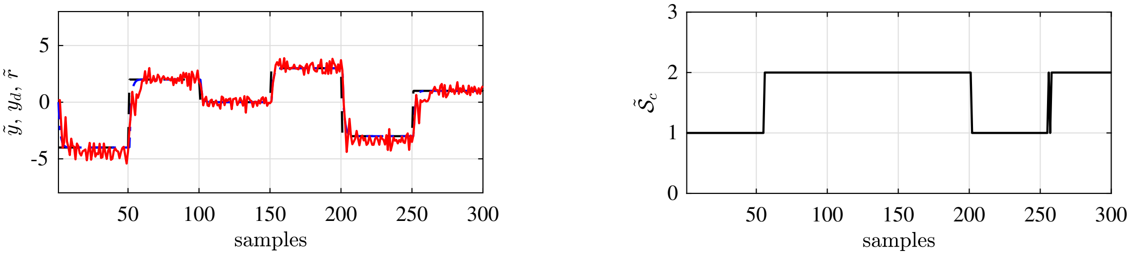

FIGURE 6 Linear time-invariant reference model: tracking a piecewise constant reference signal. [Left panel] Reference (dashed black) vs desired closed-loop behavior (dotted-dashed blue) and actual closed-loop response (red). Black, red, and blue lines are almost overlapped. [Right panel] Active mode sequence of the piecewise affine controller [Colour figure can be viewed at wileyonlinelibrary.com]

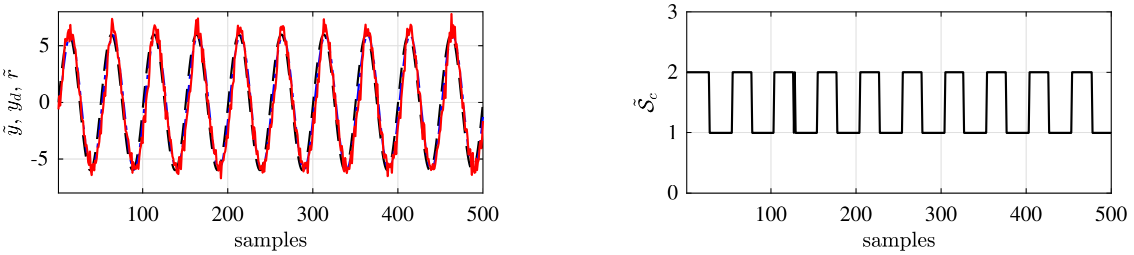

FIGURE 7 Linear time-invariant reference model: tracking a sinusoidal reference signal. [Left panel] Reference (dashed black) vs desired closed-loop behavior (dotted-dashed blue) and actual closed-loop response (red). [Right panel] Active mode sequence of the piecewise affine controller [Colour figure can be viewed at wileyonlinelibrary.com]

of the one affecting the open-loop measurements. The attained closed-loop behavior is compared with the desired one in Figure 6, where it can be noticed that the reference signal is tracked despite the measurement noise. Figure 6 further shows that switches between different local controllers are overall not frequent. The same controller is then used to track a sinusoidal reference, attaining the performance shown in Figure 7. Also, in this case, the desired and actual closed-loop responses are similar. As reported in Figure 7, the controller does not switch its state often, with its switching sequence being almost periodic, thus reflecting the characteristics of the chosen reference signal.

For the piecewise constant reference, the RMSE attained with the learned PWA controller is equal to 0.606. Instead, RMSE=0.614 is achieved with a single controller with the same structure described in (59), trained by running Step 1.1 with the same regularization parameters. When the sinusoidal reference is considered, the RMSEs attained with the PWA and LTI controllers are equal to 0.726 and 0.929, respectively. Therefore, despite the negligible difference in RMSE when considering the piecewise constant reference, the PWA controller clearly enhances the tracking performance when considering a sinusoidal reference.

6.1.1 | Sensitivity analysis

As pointed out in Section 3, the performance of the learned controller might be influenced by the chosen regularization parameters λ, λ_{aff}, and λ_p. Indeed, the selection of an inappropriate value for λ_p might negatively influence the resulting

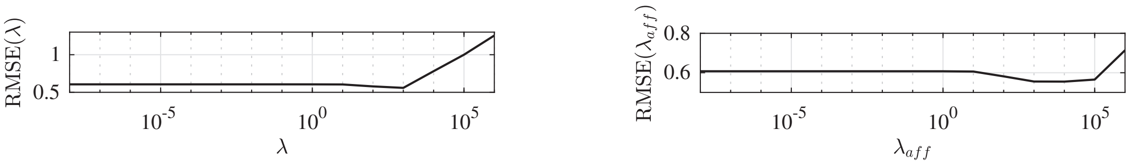

FIGURE 8 Linear time-invariant reference model: tracking a piecewise constant reference. [Left panel] Root-mean-square error (RMSE) vs λ, for $\lambda_{aff} = 10$ and $\lambda_p = 0.1$. [Right panel] RMSE vs λ_{aff}, for $\lambda = 0.1$ and $\lambda_p = 0.1$

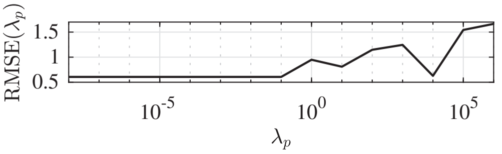

FIGURE 9 Linear time-invariant reference model: tracking a piecewise constant reference. Root-mean-square error (RMSE) vs λ_p, for $\lambda = 0.1$ and $\lambda_{aff} = 10$

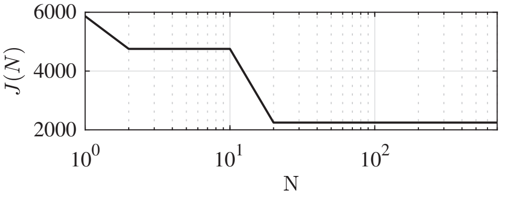

FIGURE 10 Linear time-invariant reference model: cost J vs number N of initial guesses for the mode sequence

switching logic of the controller, while improper choices of λ and λ_{aff} might deteriorate the properties of the estimated local controllers. In turn, wrong choices of these parameters might lead to poor matching of the desired closed-loop behavior, resulting in a deterioration of the tracking performance and, in the worst case, they might lead to closed-loop instability. A posterior sensitivity analysis has thus been performed with respect to each regularization parameter. Different controllers are trained for different values of λ, λ_{aff} and λ_p, by changing one tuning parameter at a time and fixing the others to the values selected in cross-validation. The performance of the resulting controllers are then evaluated in tracking the piecewise constant reference shown in Figure 6.

The RMSEs reported in Figure 8 indicate that the performance of the learned controller is fairly insensitive to the choice of λ and λ_{aff} for $\lambda, \lambda_{aff} \leq 10$, but it visibly deteriorates for $\lambda > 100$. A similar behavior is shown in Figure 9, entailing that a choice of $\lambda_p \leq 0.1$ leads to negligible differences in the performance of the resulting controller. Instead, the trained switching policy results in deteriorated closed-loop tracking performance for $\lambda_p > 0.1$.

Furthermore, the outcome of Algorithm 1 depends on the chosen initial guess for the mode sequence $\mathcal{S}_c^0$. As suggested in the introduction, to mitigate the effect of this choice the algorithm can be run for N different choices of $\mathcal{S}_c^0$, selecting the solution leading to the smallest cost. To assess the sensitivity of the method to N, Algorithm 1 is carried out for increasing values of this parameter, up to $N = 700$. As shown in Figure 10, the resulting cost and, thus, the controller does not change for $N \geq 11$.

We further evaluate the computational time required to design the controller, namely to run Algorithm 1 until one of the considered termination criteria is attained. This is performed by training a controller with data sets of increasing length, obtaining the computational times reported in Figure 11. Even for a large data set ($T = 10^5$), the CPU time required to train the controller is less than 10 seconds, proving the computational efficiency of the proposed design strategy.

We finally assess the effect of an increasing number of training samples T on the closed-loop tracking performance. As shown in Figure 12, there is no significant improvement in performance for $T \geq 3000$.

6.2 | Numerical example: PWL reference model

Let $\mathcal{G}_s$ be a PWA system with $K = 3$ possible operating conditions described by the difference equations

$$y(t) = \begin{cases} 0.4y(t-1) + 0.7u(t-1) + 0.5 & \text{if } \; y(t-1) + 6u(t-1) + 30 \leq 0, \\ 0.2y(t-1) + 0.4u(t-1) - 0.8 & \text{if } \; y(t-1) + 6u(t-1) - 30 > 0, \\ -0.3y(t-1) + 0.4u(t-1) + 0.7 & \text{otherwise.} \end{cases} \tag{60}$$

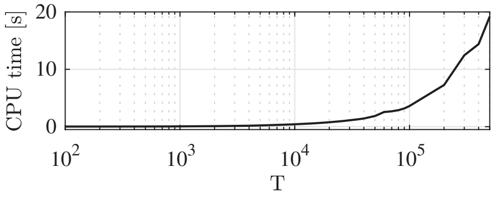

FIGURE 11 Linear time-invariant reference model: CPU time vs length of the data set $\mathcal{D}_T$. The computational time reported is the mean CPU time required to design the controller over the $N = 20$ runs of Algorithm 1

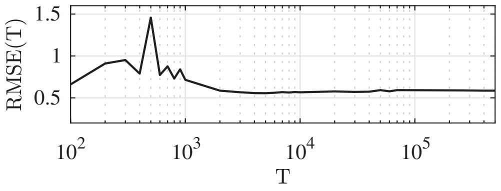

FIGURE 12 Linear time-invariant reference model: Root-mean-square error (RMSE) vs length of the data set $\mathcal{D}_T$

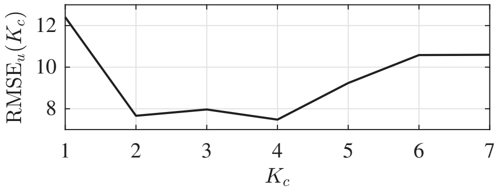

FIGURE 13 Piecewise linear reference model: RMSE_u vs number of local controllers K_c. RMSE, root-mean-square error

A data set of $T = 4750$ samples is generated by simulating the behavior of the system in open-loop, with $\{u(t)\}_{t=1}^{T}$ being a sequence of randomly generated inputs with uniform distribution in the interval $[-15, 15]$. The output is corrupted by a zero-mean Gaussian distributed noise with standard deviation $\sigma_v = 0.8$, which yields SNR=16 dB. An additional data set comprising $\bar{T} = 750$ samples is generated in similar conditions to perform cross-validation. The data used to learn the local controllers are normalized both in cross-validation and training. Differently from the previous numerical example, in this case, we aim at replicating the behavior of the following bimodal PWL reference model:

$$x_M(t+1) = \begin{bmatrix} -1 & 1 \\ -\alpha(s_M(t)) & 1 \end{bmatrix} x_M(t) + \begin{bmatrix} \alpha(s_M(t)) \\ \alpha(s_M(t)) \end{bmatrix} r(t), \tag{61a}$$

$$y_d(t) = [\ 1\ \ 0\ ] x_M(t), \tag{61b}$$

with

$$\alpha(s_M(t)) = \begin{cases} 0.7 & \text{if} - y_d(t) \leq 1 \\ 0.2 & \text{otherwise.} \end{cases} \tag{61c}$$

Nonetheless, we still design C_s so that it is composed by K_c first-order linear local controllers with the structure in (59) for all $i = 1, \ldots, K_c$. Note that, since $\mathcal{G}_s$ is forced to behave as a PWL system with partition induced by the conditions in (61c), the number of submodels cannot be easily fixed even if K is known.

The number of local controllers is set to $K_c^\star = 4$, according to the values of RMSE_u obtained in cross-validation and reported in Figure 13. As expected, the use of a single controller leads to a worse reconstruction of the input, with respect to all cases when multiple controllers are used. At the same time, the use of a relatively high number of local controllers seems to lead to a deterioration in the performance. According to the results of the cross-validation procedure, the other tuning parameters are chosen as $\lambda^\star = 10^{-4}$, $\lambda_{aff}^\star = 10^8$ and $\lambda_p^\star = 10^3$, respectively. Note that the selected value for λ_{aff} leads to $\theta_{i,3} \approx 0$ for all $i = 1, \ldots, K_c$.

Algorithm 1 is run for $N = 500$ different initial sequences and the resulting controller is tested by simulating the behavior of the interconnection between $\mathcal{G}_s$ and C_s, with the system output not corrupted by noise for better visualization. The closed-loop output obtained in tracking a piecewise constant reference are shown in Figure 14, where the reference signal is chosen to excite all possible operating conditions of $\mathcal{G}_s$. By comparing the reference, the desired and the actual closed-loop behaviors, it is clear that the attained closed-loop response follows the desired one. Indeed, according to $\mathcal{M}_s$ in (61a), the actual closed-loop system tends faster to negative references, while the transient is slower when the reference signal is positive. It is worth pointing out that the RMSE achieved with a single controller is equal to 2.247, while the one attained with the PWA controller is 0.242, clearly indicating the need for a switching controller.

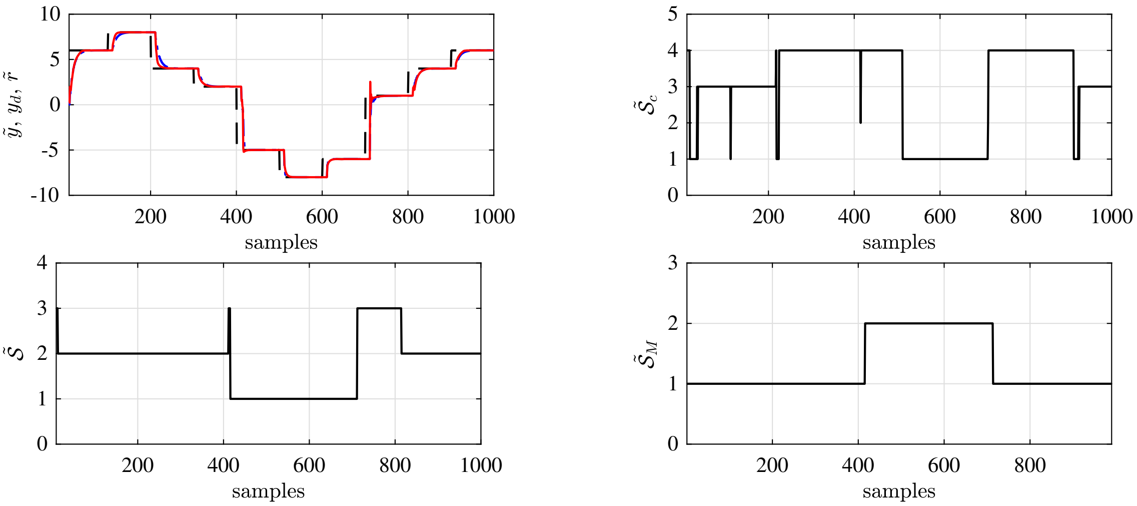

FIGURE 14 Piecewise linear reference model: tracking a piecewise constant signal. [Top left panel] Reference (dashed black) vs desired closed-loop behavior (dotted-dashed blue) and actual closed-loop response (red). Black, red, and blue lines are almost overlapped. [Top right panel] Active mode sequence of the piecewise affine controller. [Lower left panel] Discrete state of $\mathcal{G}_s$. [Lower right panel] Active mode sequence of the reference model $\mathcal{M}_s$ [Colour figure can be viewed at wileyonlinelibrary.com]

6.3 | Numerical example: a nonlinear system

To assess how a PWA controller trained with the proposed approach performs when used to control *nonlinear systems*, let $\mathcal{G}_s$ be described by the equations[62]

$$x_1(t+1) = \left(\frac{x_1(t)}{1+x_1(t)^2} + p_1\right) \sin\left(x_2(t)\right), \tag{62a}$$

$$x_2(t+1) = x_2(t)\cos x_2(t) + x_1(t) e^{-\frac{\left(x_1(t)^2+x_2(t)^2\right)}{p_2}} + \frac{u(t)^3}{1+u(t)^2+p_3\cos\left(x_1(t)+x_2(t)\right)}, \tag{62b}$$

$$y(t) = \frac{x_1(t)}{1+p_4\sin\left(x_2(t)\right)+p_5\sin\left(x_1(t)\right)} \tag{62c}$$

with parameters $p_1 = 1$ and $p_i = 0.5$, for $i = 2, \ldots, 5$. By assuming that our goal is to compensate the nonlinear behavior of the plant by closing the loop, the reference model $\mathcal{M}_s$ is chosen as the following LTI first-order discrete-time model

$$x_M(t+1) = 0.8x_M(t) + 0.2r(t), \tag{63a}$$

$$y_d(t) = x_M(t). \tag{63b}$$

A set comprising $T = 4000$ input/output pairs is constructed by performing an open-loop simulation of the system, with input chosen as the following combination of sinusoids:

$$u(t) = \sin\left(2\pi\frac{t}{25}\right) + \sin\left(2\pi\frac{t}{35}\right). \tag{64}$$

The measured output is corrupted by zero-mean white noise with Gaussian distribution and standard deviation $\sigma_v = 0.3$, so that SNR= 10.3 dB. An additional data set of length $\bar{T} = 1000$ is generated in similar conditions to perform cross-validation and select the number of local controller and the remaining tuning parameters. The values of RMSE_u obtained in cross-validation are reported in Table 1. Based on these results, the number of affine controllers is set to $K_c^\star = 3$ and, accordingly, the other tuning parameters are fixed to $\lambda^\star = 10^{-8}$, $\lambda_{aff}^\star = 10^6$ and $\lambda_p^\star = 10^{-7}$, respectively.

The controller is then designed by running Algorithm 1 with the chosen tuning parameters, for $N = 600$ possible initial guesses of the mode sequence. By considering a noisy feedback output, with the noise having the same characteristics as the one altering the open-loop measurements, the closed-loop performance attained when tracking a sinusoidal reference

TABLE 1 Artificial nonlinear system. RMSE_u vs number of local controllers K_c

	$K_c = 2$	$K_c = 3$	$K_c = 4$	$K_c = 5$	$K_c = 6$
$\text{RMSE}_u(K_c)$	0.1962	0.1961	1.978	0.2012	1.2103

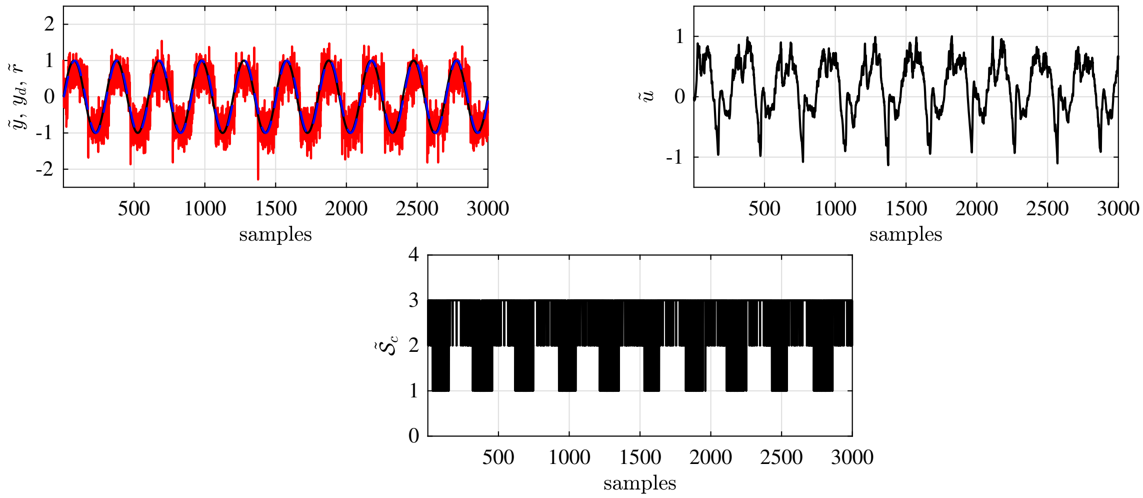

FIGURE 15 Artificial nonlinear system. [Top left panel] Sinusoidal reference (black) vs actual (red) and desired (dashed blue) closed-loop responses. [Top right panel] Input fed to the system in closed-loop. [Lower panel] Active mode sequence of the piecewise affine controller [Colour figure can be viewed at wileyonlinelibrary.com]

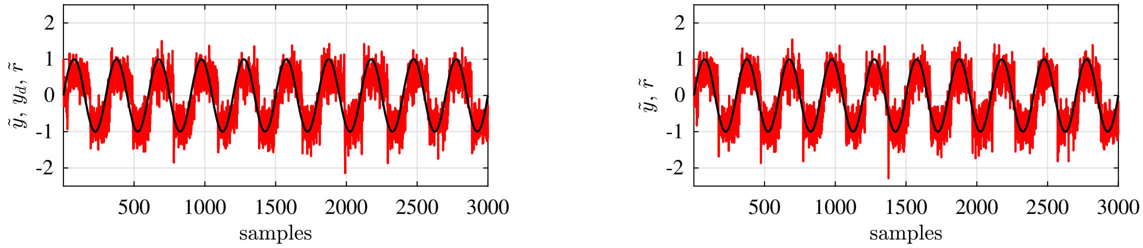

FIGURE 16 Artificial nonlinear system. [Left panel] Sinusoidal reference (black) vs closed-loop responses obtained by using a linear time-invariant (LTI) controller. [Right panel] Sinusoidal reference (black) vs closed-loop responses obtained by using a piecewise affine controller. The use of the LTI controller leads to a slight delay in tracking [Colour figure can be viewed at wileyonlinelibrary.com]

is shown in Figure 15, along with the input fed to the system and the switching sequence of the controller. Due to the nonlinearity of the system and the noise, the controller switches very frequently. Nonetheless, the reference is tracked with satisfactory performance, attaining RMSE =0.4321. Instead, a RMSE equal to 0.4800 is achieved when using a single affine controller, reflecting a delay in tracking the reference that can be noticed by looking at Figure 16. These results show that PWA controllers trained with the presented approach can be effectively used to control nonlinear systems, given the additional flexibility guaranteed by the estimation of the switching logic of the controller from data.

Note that the difference between the reconstruction errors obtained for $K_c = 2, 3, 4$ and reported in Table 1 is negligible. Therefore, controllers C_s with up to five local controllers are trained, by fixing the tuning parameters as the one leading to the values of RMSE_u reported in Table 1. By tracking the same sinusoidal reference considered in Figure 15, the RMSEs attained with the different controllers in case of uncorrupted feedback outputs are reported in Table 2. It can be clearly noticed that worse performance is attained for $K_c \leq 2$ and $K_c \geq 4$, showing that the number of local controllers chosen through cross-validation is the one leading to good closed-loop performance.

6.4 | A benchmark case study: the servo-positioning system

As a more realistic case study, we investigate the same problem considered in,[48] namely the problem of controlling the servo-positioning system with an unbalanced disc.[63] The considered system is characterized by viscous friction and the

	$K_c = 1$	$K_c = 2$	$K_c = 3$	$K_c = 4$	$K_c = 5$
RMSE(K_c)	0.3721	0.4477	0.2416	0.3164	1.2142

TABLE 2 Artificial nonlinear system. Tracking a sinusoidal reference: root-mean-square error (RMSE) vs number of local controllers K_c

TABLE 3 Physical parameters of the DC motor[63]

	Description	Values
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

presence of an additional mass that, mounted on the disk connected to the rotor, makes the mass distribution over the disk inhomogeneous. Its mathematical model is given by

$$\begin{bmatrix}\dot{\theta}(\tau)\\ \dot{\omega}(\tau)\\ \dot{I}(\tau)\end{bmatrix} = \left(\begin{bmatrix}0 & 1 & 0\\ 0 & -\frac{b}{J} & \frac{K}{J}\\ 0 & -\frac{K}{L} & -\frac{R}{L}\end{bmatrix} + \begin{bmatrix}0 & 1 & 0\\ \frac{mgl}{J} & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\frac{\sin(\theta(\tau))}{\theta(\tau)}\right)\begin{bmatrix}\theta(\tau)\\ \omega(\tau)\\ I(\tau)\end{bmatrix} + \begin{bmatrix}0\\ 0\\ \frac{1}{L}\end{bmatrix}V(\tau), \tag{65a}$$

$$y(\tau) = \theta(\tau), \tag{65b}$$

where $V(\tau)$ [V] and $I(\tau)$ [mA] are the voltage and the current over the armature, respectively, $\theta(\tau)$ [rad] is the angular position of the disc, $\omega(\tau)$ [rad/s] is its angular velocity, and the remaining parameters are reported in Table 3. It is worth pointing out that our approach can be used to tackle the considered problem, although the model in (65) is nonlinear, thanks to the universal approximation properties of PWA maps and the flexibility guaranteed by the assumption that the switching sequence of $\mathcal{G}_s$ is unknown.

As in,[48] an ideal *zero-order-hold* (ZOH) setting is considered, where the input and output sampling is synchronized with sampling period $T_s = 0.01$ seconds. The output measurements are further assumed to be filtered via an ideal anti-aliasing filter.

We build two data sets $\mathcal{D}_T$ and $\mathcal{D}_{\bar{T}}$, which comprise $T = 1500$ and $\bar{T} = 250$ samples, respectively used for training and cross-validation. Both sets are constructed by simulating the mathematical model in (65) in presence of a filtered white input voltage distributed uniformly in the interval $[-8, 8]$, with the input filter being a low-pass digital filter with cutoff frequency of 1.1 Hz. As in,[48] the output measurements are corrupted by a white zero-mean Gaussian noise sequence $v(t)$, with variance selected so that SNR=43 dB.

The reference model $\mathcal{M}_s$ is chosen as the same LTI first order discrete-time model considered in,[48] ie,

$$x_M(t+1) = 0.99x_M(t) + 0.01r(t), \tag{66a}$$

$$y_d(t) = x_M(t), \tag{66b}$$

which is characterized by a pole at about 0.15 Hz.

The structure of the controller is described as follows:

$$u(t) = -\sum_{i=1}^{8} a_i^c(s_c(t))u(t-i) + \sum_{j=0}^{8} b_j^c(s_c(t))e_{int}(t-j) + f_c(s_c(t)), \tag{67a}$$

$$e_{int}(t) = e_{int}(t-1) + (r(t) - y(t)). \tag{67b}$$

We thus aim at designing an eighth-order controller with integral action, where the latter is introduced to ensure zero steady-state error. The discrete state $s_c(t)$ of the controller is driven by the switching logic

$$s_c(t) = i \iff \Pi(t) \in \mathcal{X}_i^c, \tag{67c}$$

with

$$\Pi(t) = [\ y(t-1)\ \ y(t-2)\ \ y(t-3)\ \ y(t-4)\ ]', \tag{67d}$$

which is inspired by structure of the LPV controller trained in.[48] This means that the PWA controller changes its operating condition based on the shaft angle only, which is reasonable consided the dynamics in (65).

	$K_c = 2$	$K_c = 3$	$K_c = 4$	$K_c = 5$	$K_c = 6$
$\text{RMSE}_u(K_c)$	0.7605	0.8583	1.1229	-	-

TABLE 4 Servo-positioning system. RMSE_u vs number of local controllers K_c

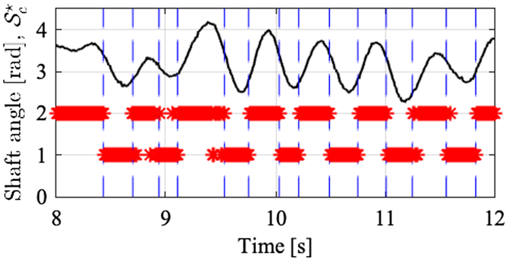

FIGURE 17 Servo-positioning system. Training set: shaft angle (black) vs reconstructed mode sequence $\mathcal{S}_c^\star$ (red stars) [Colour figure can be viewed at wileyonlinelibrary.com]

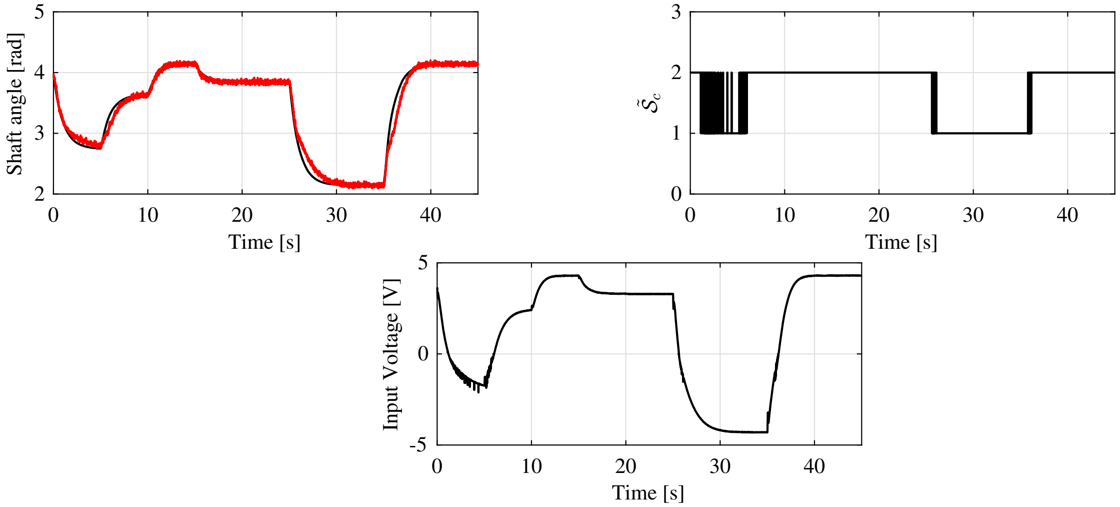

FIGURE 18 Servo-positioning system. [Top left panel] Desired closed-loop behavior (black) vs actual closed-loop response (red). [Top right panel] Active mode sequence of the piecewise affine controller. [Lower panel] Input voltage fed to the system [Colour figure can be viewed at wileyonlinelibrary.com]

The values of RMSE_u obtained in cross-validation are reported in Table 4, showing that the lowest error is attained for $K_c^\star = 2$, for $\lambda^\star = \lambda_{aff}^\star = 9 \cdot 10^5$ and $\lambda_p = 10$. The errors obtained in cross-validation for $K_c = 5$ and $K_c = 6$ are not reported in Table 4, since for these choices of K_c all the outcomes of Algorithm 1 lead to the presence of an *almost empty* cluster.[†] It is worth pointing out that the number of modes resulting from cross-validation somehow complies with the features of the system (see (65)), and, in particular, with the characteristics of its nonlinear term.

Algorithm 1 is run for $N = 100$ possible initial mode sequences $\mathcal{S}_c^0$ to learn the controller. As a result of the training phase, we obtain the reconstructed controller sequence $\mathcal{S}_c^\star$ shown in Figure 17 along with the measured angular position of the disc. The resulting switching policy clearly leads to a change in the controller when the shaft angle increases/decreases. It is interesting to observe that the estimated switching logic provides some insight into the physics of the system and, in particular, on its nonlinearity, without deriving a full mathematical model for the servo-positioning system.

The closed-loop behavior attained when the feedback output is corrupted by noise with the same characteristics of the one altering the training data is compared with the desired output in Figure 18, which further reports the switching sequence of the controller and the input fed to the system. Although as in[48] the closed-loop performance is not exactly the same for all step changes, the tracking performance is satisfactory. As shown in Figure 19 for noiseless feedback outputs, the performance attained by the PWA controller is indeed comparable with the one achieved with an LPV controller trained according to Formentin et al[48] with the PWA controller trained four times faster than the LPV controller despite

[†] A cluster is regarded as *almost empty* if it comprises less than 10% of data.

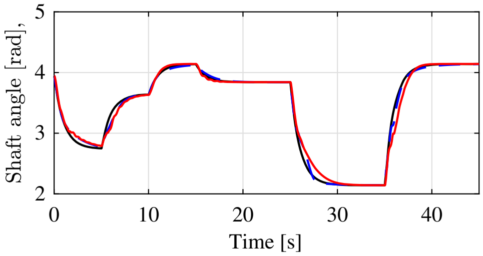

FIGURE 19 Servo-positioning system. Desired closed-loop response (black) vs closed-loop output attained with the piecewise affine controller (red) and an linear parameter-varying controller trained with the method proposed in[48] (dashed blue) [Colour figure can be viewed at wileyonlinelibrary.com]

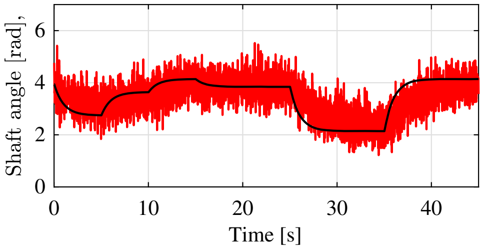

FIGURE 20 Servo-positioning system. Desired (black) vs attained (red) closed-loop response with piecewise affine controller trained for SNR~ 16 dB. SNR, signal-to-noise ratio [Colour figure can be viewed at wileyonlinelibrary.com]

the number N of possible initial conditions. Furthermore, a PWA controller can be effectively learned with data collected for SNR down to ∼ 16 dB, while efficient LPV controllers cannot be designed according to[48] for SNR lower than 43 dB, due to the quasi-LPV nature of the system. Indeed, as shown in Figure 20, the new controller still allows to track the reference signal, despite the noise corrupting the output.

All above results show that the proposed approach can be effectively used for the direct data-driven control of nonlinear systems, since the discrete state of $\mathcal{G}_s$ is assumed to be unknown.

7 | CONCLUSIONS

In this paper, we have introduced a novel approach for model-reference design of PWA controllers, which relies on IO data only and is tailored for scenarios in which the operating condition of the controlled system is unknown. By leveraging on recent results on switching system identification, the method relies on the formulation of a convex optimization problem in the design parameters. The problem is solved via a coordinate descent method alternating between the solution of the problem with respect to each optimization variable, so that both the discrete and continuous dynamics of the PWA controller are reconstructed from data. The controller is designed under the assumption that the system to be controlled is PWA. Nonetheless, this hypothesis does not restrict the effectiveness of the proposed approach to this class of systems.

The method proposed in this paper is aimed at lying the foundation for future research on direct data-driven control of hybrid systems when only IO data are available. Therefore, much work still has to be done to make the approach competitive with state-of-the-art model-based approaches. Future research in this direction will include the analysis of data-driven closed-loop stability conditions, the optimal selection of the reference model and optimal tuning of the number of local controllers and the regularization parameters.

ACKNOWLEDGEMENTS

The work of V. Breschi and S. Formentin has been partially founded by the Lombardia region and the Cariplo foundation, under the project Learning to Control (L2C),no. 2017-1520.

ORCID

Valentina Breschi https://orcid.org/0000-0002-1533-7349

REFERENCES

1. Labinaz G, Bayoumi MM, Rudie K. A survey of modeling and control of hybrid systems. *Annu Rev Control*. 1997;21:79-92.
2. Breiman L. Hinging hyperplanes for regression, classification, and function approximation. *IEEE Trans Inf Theory*. 1993;39(3):999-1013.

3. Chua L, Hasler M, Neirynck J, Verburgh P. Dynamics of a piecewise-linear resonant circuit. *IEEE Trans Circuits Syst*. 1982;29(8):535-547.
4. Heemels WPMH, De Schutter B, Bemporad A. Equivalence of hybrid dynamical models. *Automatica*. 2001;37(7):1085-1091.
5. Wicks M, DeCarlo R. Solution of coupled Lyapunov equations for the stabilization of multimodal linear systems. In: Proceedings of the 1997 American Control Conference; 1997; Albuquerque, NM.
6. Mignone D, Ferrari-Trecate G, Morari M. Stability and stabilization of piecewise affine and hybrid systems: an LMI approach. In: Proceedings of the 39th IEEE Conference on Decision and Control; 2000; Sydney, Australia.
7. Necoara I, De Schutter B, Heemels WPMH, Weiland S, Lazar M, van den Boom TJJ. Control of PWA systems using a stable receding horizon method. *IFAC Proc Vol*. 2005;38(1):123-128. 16th IFAC World Congress.
8. Rodrigues L, Boyd S. Piecewise-affine state feedback for piecewise-affine slab systems using convex optimization. *Syst Control Lett*. 2005;54(9):835-853.
9. De Schutter B, Heemels WPMH, Lunze J, Prieur C. Survey of modeling, analysis, and control of hybrid systems. In: *Handbook of Hybrid Systems Control: Theory, Tools, Applications*. Cambridge, UK: Cambridge University Press; 2009:31-56.
10. Bemporad A, Morari M. Control of systems integrating logic, dynamics, and constraints. *Automatica*. 1999;35(3):407-427.
11. Lazar M, Heemels WPMH, Weiland S, Bemporad A. Stabilizing model predictive control of hybrid systems. *IEEE Trans Autom Control*. 2006;51(11):1813-1818.
12. Istepanian R, Whidborne JF. *Digital Controller Implementation and Fragility: A Modern Perspective*. New York, NY: Springer Science & Business Media; 2012.
13. Åström KJ, Hägglund T. PID control. *IEEE Control Syst Mag*. 2006;1066(033X/06).
14. Kong J-S, Lee B-H, Kim J-G. Design of a switching PID controller using advanced genetic algorithm for a nonlinear system. In: Proceedings of the Computational Intelligence and Security, International Conference, Part I; 2005; Xi'an, China.
15. Marchetti G, Barolo M, Jovanovic L, Zisser H, Seborg DE. An improved PID switching control strategy for type 1 diabetes. *IEEE Trans Biomed Eng*. 2008;55(3):857-865.
16. Lauer F. On the complexity of piecewise affine system identification. *Automatica*. 2015;62:148-153.
17. Paoletti S, Juloski AL, Ferrari-Trecate G, Vidal R. Identification of hybrid systems: a tutorial. *Eur J Control*. 2007;13(2):242-260.
18. Garulli A, Paoletti S, Vicino A. A survey on switched and piecewise affine system identification. *IFAC Proc Vol*. 2012;45(16):344-355. Part of special issue: 16th IFAC Symposium on System Identification.
19. Ferrari-Trecate G, Muselli M, Liberati D, Morari M. A clustering technique for the identification of piecewise affine systems. *Automatica*. 2003;39(2):205-217.
20. Bemporad A, Garulli A, Paoletti S, Vicino A. A bounded-error approach to piecewise affine system identification. *IEEE Trans Autom Control*. 2005;50(10):1567-1580.
21. Juloski AL, Weiland S, Heemels WPMH. A Bayesian approach to identification of hybrid systems. *IEEE Trans Autom Control*. 2005;50(10):1520-1533.
22. Nakada H, Takaba K, Katayama T. Identification of piecewise affine systems based on statistical clustering technique. *Automatica*. 2005;41(5):905-913.
23. Breschi V, Piga D, Bemporad A. Piecewise affine regression via recursive multiple least squares and multicategory discrimination. *Automatica*. 2016;73:155-162.
24. Gosea IV, Petreczky M, Antoulas AC, Fiter C. Balanced truncation for linear switched systems. *Adv Comput Math*. 2018;44:1845-1886.
25. Shaker HR, Wisniewski R. Model reduction of switched systems based on switching generalized gramians. *Int J Innov Comput Inf Control*. 2012;8(7):5025-5044.
26. Papadopoulos AV, Prandini M. Model reduction of switched affine systems. *Automatica*. 2016;70:57-65.
27. Chen D, Bako L, Lecoeuche S. A recursive sparse learning method: application to jump Markov linear systems. *IFAC Proc Vol*. 2011;44(1):3198-3203. Part of special issue: 18th IFAC World Congress.
28. Paoletti S, Roll J, Garulli A, Vicino A. On the input-output representation of piecewise affine state space models. *IEEE Trans Autom Control*. 2010;55(1):60-73.
29. Kersting S, Buss M. Adaptive identification of continuous-time switched linear and piecewise linear systems. In: Proceedings of the 2014 European Control Conference (ECC); 2014; Strasbourg, France.
30. Formentin S, Van Heusden K, Karimi A. A comparison of model-based and data-driven controller tuning. *Int J Adapt Control Signal Process*. 2014;28(10):882-897.
31. Campi MC, Lecchini A, Savaresi SM. Virtual reference feedback tuning: a direct method for the design of feedback controllers. *Automatica*. 2002;38(8):1337-1346.
32. Formentin S, Savaresi SM, Del Re L. Non-iterative direct data-driven controller tuning for multivariable systems: theory and application. *IET Control Theory Appl*. 2012;6(9):1250-1257.
33. van Heusden K, Karimi A, Bonvin D. Data-driven model reference control with asymptotically guaranteed stability. *Int J Adapt Control Signal Process*. 2011;25(4):331-351.
34. Formentin S, Karimi A, Savaresi SM. Optimal input design for direct data-driven tuning of model-reference controllers. *Automatica*. 2013;49(6):1874-1882.
35. Formentin S, Savaresi SM. Virtual reference feedback tuning for linear parameter-varying systems. *IFAC Proc Vol*. 2011;44(1):10219-10224. Part of special issue: 18th IFAC World Congress.
36. Piga D, Formentin S, Bemporad A. Direct data-driven control of constrained systems. *IEEE Trans Control Syst Technol*. 2018;26(4):1422-1429.

37. Novara C, Formentin S, Savaresi SM, Milanese M. Data-driven design of two degree-of-freedom nonlinear controllers: the d2-IBC approach. *Automatica*. 2016;72:19-27.
38. Novara C, Formentin S. Data-driven inversion-based control of nonlinear systems with guaranteed closed-loop stability. *IEEE Trans Autom Control*. 2018;63(4):1147-1154.
39. Dai T, Sznaier M. A moments based approach to designing MIMO data driven controllers for switched systems. Paper presented at: 2018 IEEE Conference on Decision and Control (CDC); 2018; Miami Beach, FL.
40. di Bernardo M, Montanaro U, Santini S. Hybrid model reference adaptive control of piecewise affine systems. *IEEE Trans Autom Control*. 2013;58(2):304-316.
41. di Bernardo M, Montanaro U, Olm JM, Santini S. Model reference adaptive control of discrete-time piecewise linear systems. *Int J Robust Nonlinear Control*. 2013;23(7):709-730.
42. di Bernardo M, Montanaro U, Ortega R, Santini S. Extended hybrid model reference adaptive control of piecewise affine systems. *Nonlinear Anal Hybrid Syst*. 2016;21:11-21.
43. Wu C, Zhao J, Sun X. Adaptive tracking control for uncertain switched systems under asynchronous switching. *Int J Robust Nonlinear Control*. 2015;25(17):3457-3477.
44. Kersting S, Buss M. Direct and indirect model reference adaptive control for multivariable piecewise affine systems. *IEEE Trans Autom Control*. 2017;62(11):5634-5649.
45. Yuan S, De Schutter B, Baldi S. Robust adaptive tracking control of uncertain slowly switched linear systems. *Nonlinear Anal Hybrid Syst*. 2018;27:1-12.
46. Xie J, Yang D, Zhao J. Composite anti-disturbance model reference adaptive control for switched systems. *Information Sciences*. 2019;485:71-86.
47. Zhang Q, Sun X, Tong F, Chen H. A review of intelligent control algorithms applied to robot motion control. Paper presented at: 2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER); 2018; Tianjin, China.
48. Formentin S, Piga D, Tóth R, Savaresi SM. Direct learning of LPV controllers from data. *Automatica*. 2016;65:98-110.
49. Bemporad A, Breschi V, Piga D, Boyd SP. Fitting jump models. *Automatica*. 2018;96:11-21.
50. Selvi D, Piga D, Bemporad A. Towards direct data-driven model-free design of optimal controllers. Paper presented at: 2018 European Control Conference (ECC); 2018; Limassol, Cyprus.
51. Gilson M, Van den Hof P. Instrumental variable methods for closed-loop system identification. *Automatica*. 2005;41(2):241-249.
52. Petreczky M, Bako L. On the notion of persistence of excitation for linear switched systems. Paper presented at: 2011 50th IEEE Conference on Decision and Control and European Control Conference; 2011; Orlando, FL.
53. Hastie T, Tibshirani R, Friedman JH. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY: Springer; 2009. *Springer Series in Statistics*.
54. Davies DL, Bouldin DW. A cluster separation measure. *IEEE Trans Pattern Anal Mach Intell*. 1979;PAMI-1(2):224-227.
55. Formentin S, Karimi A. Enhancing statistical performance of data-driven controller tuning via l2-regularization. *Automatica*. 2014;50(5):1514-1520.
56. Vapnik VN. *Statistical Learning Theory*. Hoboken, NJ: Wiley-Interscience; 1998.
57. Rubagotti M, Zaccarian L, Bemporad A. A Lyapunov method for stability analysis of piecewise-affine systems over non-invariant domains. *Int J Control*. 2016;89(5):950-959.
58. Balkan A, Jungers RM, Kenanian J, Tabuada P. Deciding stability of a switched system without identifying it. Paper presented at: 2018 IEEE Conference on Decision and Control (CDC); 2018; Miami Beach, FL.
59. Kenanian J, Balkan A, Jungers RM, Tabuada P. Data driven stability analysis of black-box switched linear systems. arXiv preprint arXiv:1803.03687. 2018.
60. Söderström T, Stoica P. *System Identification*: Prentice-Hall; 1989.
61. Bertsekas DP. *Dynamic Programming and Optimal Control*. 2nd ed. Nashua, NH: Athena Scientific; 2000.
62. Narendra KS, Li SM. Neural networks in control systems. In: *Mathematical Perspectives on Neural Networks*. Hillsdale, NJ, USA: Lawrence Erlbaum Associates; 1996:347-394.
63. Kulcsár B, Dong J, van Wingerden J, Verhaegen M. LPV subspace identification of a DC motor with unbalanced disc. *IFAC Proc Vol*. 2009;42(10):856-861. Part of special issue: 15th IFAC Symposium on System Identification.

How to cite this article: Breschi V, Formentin S. Direct data-driven design of switching controllers. *Int J Robust Nonlinear Control*. 2019;1–31. https://doi.org/10.1002/rnc.4821

APPENDIX

PROOF OF THEOREM 1

1. Consider the problem of optimizing $\mathcal{J}_1$ defined in (26b) for a fixed mode sequence, ie,

$$\min_{\Theta} \sum_{t=1}^{T} \left(X_c(\xi(t))'\theta_{s_c^{k-1}(t)} - u(t) \right)^2 + \lambda \|\Theta\|_2^2. \tag{A1}$$

Thanks to the separability of the objective function, the parameters of each local controller can be estimated by solving K_c distinct regularized least-squares problems, namely,

$$\theta_i^k = \arg\min_{\Theta} \sum_{t:s_c^{k-1}(t)=i} \left(X_c(\xi(t))'\theta_{s_c^{k-1}(t)} - u(t) \right)^2 + \lambda \|\theta_i\|^2, \quad i = 1, \dots, K_c. \tag{A2}$$

We prove the polarization of the estimates obtained by solving problem (A2) in an *ideal* setting. Indeed, we assume that the polyhedral partition $\{\mathcal{X}_{o,i}^c\}_{i=1}^{K_c}$ and the sequence $\mathcal{S}_{o,c}$ satisfying Assumptions 1-3 are known. Problem (A2) is thus solved by imposing $s_c^{k-1}(t) = s_{o,c}(t)$, for all $t \in \mathcal{I}_1^T$.

Consider the closed-form solution of Problem (A2) given by

$$\theta_i^k = \left(\frac{1}{c_i} \sum_{t:s_{o,c}(t)=i} X_c(\xi(t))X_c(\xi(t))' + \frac{\lambda}{c_i} I_{n_c} \right)^{-1} \left(\frac{1}{c_i} \sum_{t:s_{o,c}(t)=i} X_c(\xi(t))u(t) \right), \tag{A3}$$

with c_i being the number of data points associated to i-th local controller, according to the mode sequence $\mathcal{S}_{o,c}$. Under Assumption 1, the solution $\theta_{o,i}$ of Problem (12) satisfies the condition

$$u(t) = X_c(\xi_o(t))'\theta_{o,i}, \ \ \forall t \, : \, s_{o,c}(t) = i, \tag{A4}$$

or, equivalently,

$$u(t) = X_c(\xi(t))'\theta_{o,i} + \Delta X_c(\xi(t))'\theta_{o,i}, \ \ \forall t : \ \ s_{o,c}(t) = i, \tag{A5}$$

with $\Delta X_c(\xi(t)) = X_c(\xi_o(t)) - X_c(\xi(t))$. Notice that

$$\Delta X_c(\xi(t)) = \begin{bmatrix} 0 & \dots & 0 & \xi_o(t) - \xi(t) & \dots \xi_o\left(t - n_b^c\right) - \xi\left(t - n_b^c\right) \end{bmatrix}' \tag{A6}$$

and $\xi_o(t) - \xi(t) = (M^\dagger(s_M(t)) - 1)v(t)$.

When c_i tends to infinity (and thus the dimension T of the data set is such that $T \to \infty$), the expression in (A3) can be simplified as follows:

$$\theta_i^k = \left(\frac{1}{c_i} \sum_{t:s_{o,c}(t)=i} X_c(\xi(t))X_c(\xi(t))' \right)^{-1} \left(\frac{1}{c_i} \sum_{t:s_{o,c}(t)=i} X_c(\xi(t))u(t) \right), \tag{A7}$$

since λ is finite. By substituting (A5) into (A7), it is straightforward to prove that the solution of problem (A2) is equal to

$$\theta_i^k = \theta_i^o + \left(\frac{1}{c_i} \sum_{t:s_{o,c}(t)=i} X_c(\xi(t))X_c(\xi(t))' \right)^{-1} \left(\frac{1}{c_i} \sum_{t:s_{o,c}(t)=i} X_c(\xi(t))\Delta X_c(\xi(t))'\theta_i^o \right). \tag{A8}$$

Consider the expression for the measured output in (9) and the definitions of $X_c(\xi(t))$ and $\Delta X_c(\xi(t))$ in (19) and (A6), respectively. It is clear that $X_c(\xi(t))$ and $\Delta X_c(\xi(t))'$ are correlated even when the noise $v(t)$ is white. Therefore, for

$c_i \to \infty$, the second term on the right-hand-side of (A8) is nonzero, thus proving the polarization of the solution of problem (25).

2. Under Assumptions 1-3, the solution $\Theta_{\rm o}$ of Problem (12) satisfies the condition

$$U = R_{\rm o}(\xi_{\rm o})'\Theta_{\rm o}, \tag{A9}$$

with $R_{\rm o}(\xi_{\rm o})$ being a matrix in $\xi_{\rm o}(t)$ (32) and in the ideal mode sequence $\mathcal{S}_{{\rm o},c}$, given by

$$R_{\rm o}(\xi_{\rm o}) = \begin{bmatrix} X_c(\xi_{\rm o}(1))'\mathbb{1}_{[s_{{\rm o},c}(1)]} \\ \vdots \\ X_c(\xi_{\rm o}(T))'\mathbb{1}_{[s_{{\rm o},c}(T)]} \end{bmatrix}. \tag{A10}$$

Condition (A9) can be equivalently written as

$$U = R(\xi)'\Theta_{\rm o} + \underbrace{(R(\xi_{\rm o}) - R(\xi))'}_{\Delta X_c(\xi)}\Theta_{\rm o} + \underbrace{(R_{\rm o}(\xi_{\rm o}) - R(\xi_{\rm o}))'}_{\Delta X_c(\xi_{\rm o})}\Theta_{\rm o}, \tag{A11}$$

where $\Delta X_c(\xi)$ accounts for the differences between $\xi(t)$ and $\xi_{\rm o}(t)$, while $\Delta X_c(\xi_{\rm o})$ depends on the mismatch between $\mathcal{S}_c^{k-1}$ and $\mathcal{S}_{{\rm o},c}$. Consider now the equivalent expression for the closed-form solution of problem (37)

$$\Theta^k = \left(\frac{1}{T}(Z'R(\xi))'Z'R(\xi) + \frac{\lambda}{T}I_{n_\Theta}\right)^{-1}\frac{1}{T}(Z'R(\xi))'Z'U, \tag{A12}$$

obtained by dividing and multiplying (38a) for the length T of the available data set. When $T \to \infty$, Equation (A12) can be simplified as follows:

$$\lim_{T\to\infty} \Theta^k = \lim_{T\to\infty}\left(\frac{1}{T}Z'R(\xi)\right)^{-1}\frac{1}{T}Z'U, \tag{A13}$$

since λ is finite. Then, by replacing the input vector U in (A13) with its expression in (A11), the limit in (A13) is equivalent to

$$\lim_{T\to\infty} \Theta^k = \Theta^o + \lim_{T\to\infty}\left(\frac{1}{T}Z'R(\xi)\right)^{-1}\frac{1}{T}Z'\Delta X_c(\xi)\Theta^o + \lim_{T\to\infty}\left(\frac{1}{T}Z'R(\xi)\right)^{-1}\frac{1}{T}Z'\Delta X_c(\xi_{\rm o})\Theta^o. \tag{A14}$$

According to Assumption 6, it holds that

$$\lim_{T\to\infty}\left(\frac{1}{T}Z'R(\xi)\right)^{-1} < \infty, \tag{A15}$$

and, thus, to prove the consistency of Θ^k, we have to show that the following equalities are verified:

$$\lim_{T\to\infty}\frac{1}{T}Z'\Delta X_c(\xi) = 0, \tag{A16a}$$

$$\lim_{T\to\infty}\frac{1}{T}Z'\Delta X_c(\xi_{\rm o}) = 0. \tag{A16b}$$

By definition of Z and $R(\xi)$ in (36), it holds that

$$Z'\Delta X_c(\xi) = \sum_{t=1}^{T}\zeta(t)\left[0 \ \dots \ 0 \ \xi_{\rm o}(t) - \xi(t) \ \dots \ \xi_{\rm o}\left(t - n_b^c\right) - \xi\left(t - n_b^c\right)\right]\mathbb{1}_{[s_c^{k-1}(t)]}, \tag{A17}$$

and thus, under Assumption 6, the equality in (A16a) is verified.

It additionally holds that

$$Z'\Delta X_c(\xi_{\rm o}) = \sum_{t=1}^{T}\zeta(t)X_c(\xi_{\rm o}(t))\left(\mathbb{1}_{[s_{{\rm o},c}(t)]} - \mathbb{1}_{[s_c^{k-1}(t)]}\right) = \sum_{t\in M}\zeta(t)X_c(\xi_{\rm o}(t))\left(\mathbb{1}_{[s_{{\rm o},c}(t)]} - \mathbb{1}_{[s_c^{k-1}(t)]}\right), \tag{A18}$$

where all the terms not associated to samples in M, with M defined as in Assumption 5, have been removed, as the only elements in the sum that are nonzero are the ones associated to the misclassified points. The limit that has to be studied is thus the following:

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t\in M} \zeta(t) X_c(\xi_{\mathrm{o}}(t)) \left(\mathbb{1}_{[s_{\mathrm{o},c}(t)]} - \mathbb{1}_{[s_c^{k-1}(t)]} \right). \tag{A19}$$

By dividing and multiplying (A19) for the cardinality $\#m$ of M, we obtain that

$$\lim_{T\to\infty} \frac{\#m}{T} \left(\frac{1}{\#m} \sum_{t\in M} \zeta(t) X_c(\xi_{\mathrm{o}}(t)) \mathbb{1}_{[s_{\mathrm{o},c}(t)]} - \frac{1}{\#m} \sum_{t\in M} \zeta(t) X_c(\xi_{\mathrm{o}}(t)) \mathbb{1}_{[s_c^{k-1}(t)]} \right). \tag{A20}$$

Because of Assumption 6, it holds that

$$\frac{1}{\#m} \sum_{t\in M} \zeta(t) X_c(\xi_{\mathrm{o}}(t)) \mathbb{1}_{[s_{\mathrm{o},c}(t)]} < \infty, \tag{A21a}$$

$$\frac{1}{\#m} \sum_{t\in M} \zeta(t) X_c(\xi_{\mathrm{o}}(t)) \mathbb{1}_{[s_{\mathrm{o},c}(t)]} < \infty, \tag{A21b}$$

and, since $\#m \ll T$ (see Assumption 5), the equality in (A16b) holds. The consistency result in (40) straightforwardly follows.