Observability and Controllability of Piecewise Affine and Hybrid Systems

Alberto Bemporad, Giancarlo Ferrari-Trecate, and Manfred Morari

***Abstract*—In this paper, we prove, in a constructive way, the equivalence between piecewise affine systems and a broad class of hybrid systems described by interacting linear dynamics, automata, and propositional logic. By focusing our investigation on the former class, we show through counterexamples that observability and controllability properties cannot be easily deduced from those of the component linear subsystems. Instead, we propose practical numerical tests based on mixed-integer linear programming.**

***Index Terms*—Controllability, hybrid systems, mixed-integer linear programming, observability, piecewise affine systems, piecewise linear systems.**

I. INTRODUCTION

IN recent years, both control and computer science have been attracted by *hybrid systems* [1], [2], [23], [25], [26] because they provide a unified framework for describing processes evolving according to continuous dynamics, discrete dynamics, and logic rules. The interest is mainly motivated by the large variety of practical situations, for instance, real-time systems, where physical processes interact with digital controllers.

Several modeling formalisms have been developed to describe hybrid systems, as reviewed in [24]. It is apparent that the tools for the analysis of hybrid systems strongly depend on the adopted mathematical description. Computer scientists have extended automata theory to *timed automata*, where the continuous-time flow is modeled as $\dot{x} = 1$, and further to *linear hybrid automata* [1], where the dynamic is specified by the differential inclusion $a \leq \dot{x} \leq b$. On the other side, the control community started studying the so-called *hybrid dynamical systems* [11] or *hybrid automata* [26] where the switching between different dynamics is governed by a finite automaton. A special case where dynamic equations and switching rules are linear functions of the state are the so-called piecewise affine (PWA) systems [33].

Recently, Bemporad and Morari [4] introduced a new class of hybrid systems called mixed logical dynamical (MLD) systems. The justification for the MLD form is that it is capable of modeling a broad class of systems arising in many applications: linear hybrid dynamical systems, hybrid automata, nonlinear dynamic systems where the nonlinearity can be approximated by a piecewise linear function, some classes of discrete event systems, linear systems with constraints, etc. Examples of real-world applications that can be naturally modeled within the MLD framework are reported in [3]–[5].

MLD systems are formulated in discrete time. Although the effects of sampling can be neglected in most applications, subtle phenomena such as Zeno behaviors cannot be captured in discrete time. On the other hand, although reformulating MLD systems in continuous time would be quite easy from a theoretical point of view, a discrete-time formulation allows developing numerically tractable schemes for solving complex problems, such as control [4], state estimation and fault detection [3], [15], and formal verification of hybrid systems [5], [6]. For this reason, the analysis presented in this paper will be limited to discrete time.

The first result is to prove, in a constructive way, that MLD systems are formally equivalent to PWA systems. This result allows extending all of the techniques developed for PWA models to the general MLD description of hybrid systems, therefore rendering the PWA framework a useful companion for investigating properties and designing algorithms. Although based on different arguments, this importance has also been pointed out by Sontag [33], who highlights the equivalence between piecewise linear (PWL) systems and interconnections of linear systems and finite automata.

Piecewise affine systems are described by the state-space equations

$$\begin{aligned} x(t+1) &= A_i x(t) + B_i u(t) + f_i \\ y(t) &= C_i x(t) + g_i \end{aligned} \qquad \text{for } \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \in \mathcal{X}_i \tag{1}$$

where $\{\mathcal{X}_i\}_{i=1}^s$ is a partition of the state+input set and f_i, g_i are suitable constant vectors. Each subsystem defined by the 5-tuple $(A_i, B_i, f_i, C_i, g_i)$, $i \in \{1, 2, \cdots, s\}$ is termed a *component* of the PWA system (1). If f_i and g_i are null, system (1) is referred to as piecewise linear. From a complexity point of view, PWL and PWA systems are equivalent (f_i, g_i can be thought of as generated by integrators with no input).

PWA systems are sufficiently expressive to model a large number of physical processes, such as systems with static nonlinearities (for instance, actuator saturation), and they can approximate nonlinear dynamics with arbitrary accuracy via multiple linearizations at different operating points.

Despite the fact that PWA models are just a composition of linear time-invariant dynamic systems, their structural properties such as observability, controllability, and stability are complex and articulated, as is typical of nonlinear systems.

Manuscript received June 1, 1999; revised December 3, 1999. Recommended by Associate Editor, S. Weiland. This work was supported by the Swiss National Science Foundation.

The authors are with the Institut für Automatik, ETH—Swiss Federal Institute of Technology, CH 8092 Zürich, Switzerland (e-mail: bemporad, ferrari, morari@aut.ee.ethz.ch).

Publisher Item Identifier S 0018-9286(00)09445-9.

0018–9286/00$10.00 © 2000 IEEE

Consider, for instance, stability properties. Besides simple, but very conservative results such as finding one common quadratic Lyapunov function for all of the components, researchers have started developing analysis and synthesis tools for PWA systems only very recently. By adopting piecewise quadratic Lyapunov functions, a computational approach based on linear matrix inequalities has been proposed in [19] and [21] for stability analysis and control synthesis. Construction of Lyapunov functions for switched systems has also been tackled in [37]. For the general class of switched systems of the form $\dot{x} = f_i(x), i = 1, \cdots, s$, an extension of the Lyapunov criterion based on multiple Lyapunov functions was introduced in [9] and [10]. Blondel and Tsitsiklis [8] showed that the stability of autonomous PWL systems is $\mathcal{NP}$ hard to verify (i.e., in general, the stability of a PWL system cannot be assessed by a polynomial-time algorithm, unless $\mathcal{P} = \mathcal{NP}$), even in the simple case of two component subsystems. Several global properties (such as global convergence and asymptotic stability) of PWA systems have been recently shown undecidable in [7].

The research into stability criteria for PWL systems has been motivated by the fact that the stability of each component subsystem is not enough to guarantee stability of a PWL system (and vice versa). Branicky [10] gives an example where stable subsystems are suitably combined to generate an unstable PWL system. Stable systems constructed from unstable ones have been reported in [36]. These examples point out that restrictions on the switching have to be imposed in order to prove that a PWL composition of stable components remains stable.

Very little research focused on the observability and controllability properties of hybrid systems, apart from contributions limited to the field of timed automata [1], [20], [23] and the pioneering work of Sontag [30] for PWL systems. Needless to say, these concepts are fundamental for understanding *if* and *how well* a state observer and a controller for a hybrid system can be designed. For instance, observability properties were directly exploited for designing convergent state estimation schemes for hybrid systems in [15].

Controllability and observability properties have been investigated in [14] and [18] for linear time-varying systems, and in particular for the so-called class of piecewise constant systems (where the matrices in the state-space representation are piecewise constant functions of time). Although in principle applicable, these results do not allow one to catch the peculiarities of PWA systems.

General questions of the $\mathcal{NP}$ hardness of the controllability of nonlinear systems were addressed by Sontag [32]. Following his earlier results [30], [31], Sontag [33] analyzes the computational complexity of the observability and controllability of PWA systems through arguments based on the language of piecewise linear algebra. The author proves that observability/controllability is $\mathcal{NP}$ complete over finite time, and is undecidable over infinite time (i.e., in general, cannot be solved in finite time by means of any algorithm). Using a different rationale, the same result was derived in [8].

In this paper, we provide two main contributions to the analysis of the controllability and observability of hybrid and PWA systems: 1) we show the reader that observability and controllability properties can be very complex; we present a number of counterexamples that rule out obvious conjectures about inheriting observability/controllability properties from the composing linear subsystems[1]; and 2) we provide observability and controllability tests based on *linear* and *mixed-integer linear programs* (MILP).

II. Mixed Logical Dynamical (MLD) Systems

The mixed logical dynamical (MLD) form was introduced in [4], based on the idea of transforming logic relations into mixed-integer linear inequalities [28], [35]. It is a modeling framework that allows the description of various classes of systems, like systems with mixed discrete/continuous inputs and states, automata driven by events on continuous dynamics, systems with qualitative outputs, and PWA systems. The ability to include constraints, constraint prioritization, and heuristics augments the expressiveness and generality of the MLD framework. The general MLD form is

$$x(t+1) = Ax(t) + B_1u(t) + B_2\delta(t) + B_3z(t) \quad (2a)$$

$$y(t) = Cx(t) + D_1u(t) + D_2\delta(t) + D_3z(t) \quad (2b)$$

$$E_2\delta(t) + E_3z(t) \leq E_1u(t) + E_4x(t) + E_5 \quad (2c)$$

where $x \in \mathbb{R}^{n_c} \times \{0,1\}^{n_\ell}$ are the continuous and binary states, $u \in \mathbb{R}^{m_c} \times \{0,1\}^{m_\ell}$ are the inputs, $y \in \mathbb{R}^{p_c} \times \{0,1\}^{p_\ell}$ are the outputs, and $\delta \in \{0,1\}^{r_\ell}$, $z \in \mathbb{R}^{r_c}$ represent auxiliary binary and continuous variables, respectively. All constraints on state, input, z, and δ variables are summarized in the inequality (2c). Although the description (2) seems to be linear, nonlinearity is concentrated and hidden in the integrality constraints over binary variables.

We assume that system (2) is *completely well posed* [4], which in words means that, for all x, u within a bounded set, the variables δ, z are uniquely determined, i.e., there exist functions F, G such that, at each time t, $\delta(t) = F(x(t), u(t))$, $z(t) = G(x(t), u(t))$.[2] This allows assuming that $x(t+1)$ and $y(t)$ are uniquely defined once $x(t)$, $u(t)$ are given, and therefore that x and y trajectories exist and are uniquely determined by the initial state $x(0)$ and input trajectory u.

The auxiliary variables are introduced when transforming propositional logic into linear inequalities. We briefly review here these translation techniques, and refer the reader to [4] for a detailed exposition.

By following standard notation [12], [35], [38], [39], we adopt capital letters X_i to represent statements, e.g., "$x \geq 0$" or "temperature is hot." X_i is commonly referred to as a *literal*, and has a *truth value* of either "T" (true) or "F" (false). Boolean algebra enables statements to be combined in compound statements by means of *connectives:* "$\wedge$" (and), "$\vee$" (or), "$\sim$" (not), "$\rightarrow$" (implies), "$\leftrightarrow$" (if and only if), "$\oplus$" (exclusive or). Connectives satisfy several properties (see, e.g., [13]), which can be used to transform compound statements

[1] We thank one of the anonymous reviewers for pointing out that similar counterexamples were orally presented by Leonid Gurvits.

[2] A more general definition of well posedness, where only the components of δ and z entering (2a)–(2b) are required to be unique, is given in [4].

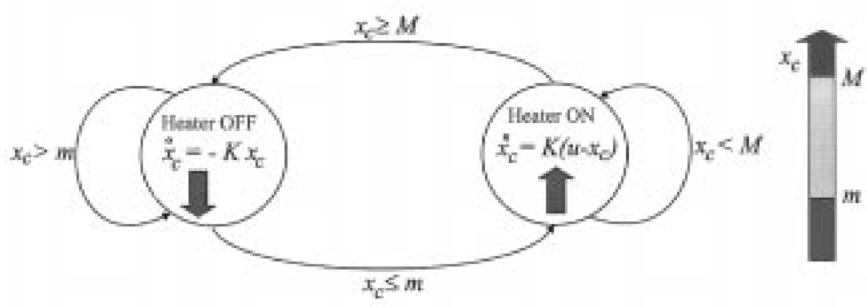

Fig. 1. Temperature control system.

TABLE I
BASIC CONVERSION OF LOGIC RELATIONS INTO MIXED-INTEGER INEQUALITIES; RELATIONS INVOLVING THE FORM $[\delta = 0]$ CAN BE OBTAINED BY SUBSTITUTING $(1-\delta)$ FOR δ IN THE CORRESPONDING INEQUALITIES

	relation	logic	mixed integer (in)equalities
P1	AND ($\wedge$)	$[\delta_1 = 1] \wedge [\delta_2 = 1]$	$\delta_1 = 1$ $\delta_2 = 1$
P2		$[\delta_3 = 1] \leftrightarrow$ $[\delta_1 = 1] \wedge [\delta_2 = 1]$	$-\delta_1 + \delta_3 \le 0$ $-\delta_2 + \delta_3 \le 0$ $\delta_1 + \delta_2 - \delta_3 \le 1$
P3	OR ($\vee$)	$[\delta_1 = 1] \vee [\delta_2 = 1]$	$\delta_1 + \delta_2 \ge 1$
P4		$[\delta_3 = 1] \leftrightarrow$ $[\delta_1 = 1] \vee [\delta_2 = 1]$	$\delta_1 - \delta_3 \le 0$ $\delta_2 - \delta_3 \le 0$ $-\delta_1 - \delta_2 + \delta_3 \le 0$
P5	NOT ($\sim$)	$\sim [\delta_1 = 1]$	$\delta_1 = 0$
P6	XOR ($\oplus$)	$[\delta_1 = 1] \oplus [\delta_2 = 1]$	$\delta_1 + \delta_2 = 1$
P7		$[\delta_3 = 1] \leftrightarrow$ $[\delta_1 = 1] \oplus [\delta_2 = 1]$	$-\delta_1 - \delta_2 + \delta_3 \le 0$ $-\delta_1 + \delta_2 - \delta_3 \le 0$ $\delta_1 - \delta_2 - \delta_3 \le 0$ $\delta_1 + \delta_2 + \delta_3 \le 2$
P8	IMPLY ($\rightarrow$)	$[\delta_1 = 1] \rightarrow [\delta_2 = 1]$	$\delta_1 - \delta_2 \le 0$
P9		$[f(x) \le 0] \rightarrow [\delta = 1]$	$f(x) \ge \epsilon + (m - \epsilon)\delta$
P10		$[\delta = 1] \rightarrow [f(x) \le 0]$	$f(x) \le M - M\delta$
P11	IFF ($\leftrightarrow$)	$[\delta_1 = 1] \leftrightarrow [\delta_2 = 1]$	$\delta_1 - \delta_2 = 0$
P12		$[f(x) \le 0] \leftrightarrow [\delta = 1]$	$f(x) \le M - M\delta$ $f(x) \ge \epsilon + (m - \epsilon)\delta$
P13	Product	$z = \delta \cdot f(x)$	$z \le M\,\delta$ $-z \le -m\,\delta$ $z \le f(x) - m(1-\delta)$ $-z \le -f(x) + M(1-\delta)$

into equivalent statements involving different connectives, and simplify complex statements. Correspondingly, one can associate with a literal X_i a *logical variable* $\delta_i \in \{0, 1\}$, which has a value of either 1 if X_i = T, or 0 otherwise. A propositional logic problem, where a statement X_1 must be proved to be true given a set of (compound) statements involving literals $X_1, \cdots, X_n$ can be solved by means of a linear integer program by suitably translating the original compound statements into linear inequalities involving logical variables δ_i. In fact, the propositions and linear constraints reported in Table I can easily be seen to be equivalent.

These translation techniques can be adopted to model logical parts of processes and heuristic knowledge about plant operation as integer linear inequalities. The link between logic statements and continuous dynamical variables, in the form of logic statements derived from conditions on physical dynamics, is provided by properties (P9)–(P12) in Table I, and leads to *mixed-integer linear inequalities*, i.e., linear inequalities involving both *continuous variables* of $\mathbb{R}^n$ and logical (*indicator*) variables in $\{0, 1\}$. Consider, for instance, the statement $X \triangleq [f(x) \le 0]$ where $f\colon \mathbb{R}^n \mapsto \mathbb{R}$ is linear, assume that $x \in \mathcal{X}$, where $\mathcal{X} \subset \mathbb{R}^n$ is a given bounded set, and define

$$M \triangleq \max_{x \in \mathcal{X}} f(x), \quad m \triangleq \min_{x \in \mathcal{X}} f(x).$$

Theoretically, an over[under]estimate of M $[m]$ suffices for our purpose. By associating a binary variable δ with the literal X, one can transform $X \triangleq [f(x) \le 0]$ into mixed-integer inequalities as described in (P12), Table I, where ϵ is a small tolerance (typically the machine precision), beyond which the constraint is regarded as violated. Note that, sometimes, translations require the introduction of *auxiliary variables* [39, p. 178]; for instance, according to (P13), a product between logic and continuous quantities requires the introduction of a real variable z.

The rules of Table I can be generalized for relations involving an arbitrary number of discrete variables combined by arbitrary connectives. *Any* combinational relation of logical variables can be, in fact, represented in conjunctive normal form (CNF), and subsequently automatically translated (without using additional integer variables) into mixed-integer linear inequalities. This requires the translation from the original logic statement to CNF. An alternative method for translating any logical relation between Boolean literals, given in the form of a logical proposition or truth table, into a minimal set of linear integer inequalities has been recently shown in [27].

In light of the transformations of Table I, it is clear that the well-posedness assumption stated above is usually guaranteed by the way the linear inequalities (2c) are generated, and therefore this hypothesis is typically verified by MLD relations derived from modeling real-world plants. Nevertheless, a numerical test for well-posedness is reported in [4, Appendix 1].

A. An Example: Temperature Control System

In order to exemplify the modeling techniques of MLD systems, we consider the temperature controller example reported in [1]. The temperature x_c of a room is controlled through a thermostat, which turns a heater on and off according to the measured temperature. When the heater is off, x_c decreases according to the first-order dynamics $\dot{x}_c = -Kx_c$; when the heater is on, $\dot{x}_c = K(u - x_c)$, where u is proportional to the power of the heater, $m_u \le u \le M_u$. While in [1] u is considered constant, here we allow more degrees of freedom by assuming that u is an exogenous input. The hybrid automaton modeling the temperature control system is depicted in Fig. 1. In order to translate the automaton into the MLD form (2), we discretize the continuous dynamics with sampling time T_s, namely,

$$x_c(t+1) = \begin{cases} \lambda x_c(t), & \text{if heater OFF} \\ \lambda x_c(t) + (1-\lambda)u(t), & \text{if heater ON} \end{cases} \tag{3}$$

where $\lambda \triangleq e^{-KT_s}$. Then, we introduce the auxiliary binary variables

$$[\delta_1(t) = 1] \leftrightarrow [x_c(t) \geq M] \tag{4a}$$
$$[\delta_2(t) = 1] \leftrightarrow [x_c(t) \leq m] \tag{4b}$$

which take into account the crossing of the guard lines (obviously, $m < M$). Equations (4a)–(4b) can be transformed into mixed-integer linear inequalities by using (P12) in Table I (we assume that a lower bound m_x and an upper bound M_x over x_c are known).

A logic state $x_\ell(t)$ is needed to store the status of the heater, and evolves according to the equation

$$x_\ell(t+1) = \delta_3(t) \tag{5}$$

where

$$[\delta_1(t) = 1] \rightarrow [\delta_3(t) = 0] \tag{6a}$$
$$[\delta_2(t) = 1] \rightarrow [\delta_3(t) = 1] \tag{6b}$$
$$[\delta_1(t) = 0] \wedge [\delta_2(t) = 0] \rightarrow [\delta_3(t) = \delta_4(t)] \tag{6c}$$

and

$$\delta_4(t) = x_\ell(t) \tag{7}$$

(although δ_4 is redundant here, the reason for introducing it will be clear in Section III).

As δ_1 and δ_2 cannot be 1 at the same time, we include the constraint

$$\delta_1(t) + \delta_2(t) \leq 1. \tag{8}$$

Equations (6a) and (6b) are translated into inequalities according to (P8). Equation (6c) is equivalent to

$$\delta_1(t) + \delta_2(t) \leq \delta_3(t) - \delta_4(t) \leq \delta_1(t) + \delta_2(t). \tag{9}$$

Although (9) can be immediately verified by inspection, it has been obtained by applying the technique described in [27] to transform general propositional logic statements into mixed-integer linear inequalities through polyhedral computation.

The dynamics (3) can be equivalently rewritten as

$$x_c(t+1) = \lambda x_c(t) + x_\ell(t)\,(1-\lambda)u(t). \tag{10}$$

Because of the product involving $x_\ell(t)$ and $u(t)$, we introduce the auxiliary continuous variable $z(t) = x_\ell(t)u(t) = \delta_4(t)u(t)$, which can be transformed into mixed-integer linear inequalities, according to (P13) in Table I.

The transformations above can be summarized in the following MLD representation of the temperature control system:

$$x(t+1) = \begin{bmatrix} \lambda & 0 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \delta(t) + \begin{bmatrix} 1-\lambda \\ 0 \end{bmatrix} z(t) \tag{11a}$$

$$\begin{bmatrix} 0 & M_x - m & 0 & 0 \\ 0 & m_x - m - \epsilon & 0 & 0 \\ M - m_x & 0 & 0 & 0 \\ M - M_x - \epsilon & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 1 & 0 \\ -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -M_u \\ 0 & 0 & 0 & m_u \\ 0 & 0 & 0 & -m_u \\ 0 & 0 & 0 & M_u \end{bmatrix} \delta + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} z \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ -1 \end{bmatrix} u + \begin{bmatrix} -1 & 0 \\ 1 & 0 \\ 1 & 0 \\ -1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} M_x \\ -m - \epsilon \\ -m_x \\ M - \epsilon \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -m_u \\ M_u \end{bmatrix} \tag{11b}$$

where $\delta \triangleq [\delta_1\ \delta_2\ \delta_3\ \delta_4]'$ and $x \triangleq [x_c\ x_\ell]'$. A simulation of the system for $T_s = 0.1$, $K = 1$, $M = 20$, $m = 10$, $M_u = 30$, $m_u = 1$, $M_x = 100$, $m_x = 0$, starting from the initial condition $x_c(0) = M$, $x_\ell(0) = 0$, and applying the input $u(t) \equiv 24$ is depicted in Fig. 2.

This example has shown the main steps to represent a hybrid system in the MLD form (2). This procedure was recently automatized by the language HYSDEL (hybrid system description language), developed at ETH Zürich. The HYSDEL compiler automatically generates the matrices of the MLD system starting from a high-level description of the hybrid system, and is available at http://control.ethz.ch/~hybrid/hysdel.

III. Equivalence between Hybrid and PWA Systems

Consider a *piecewise affine* (PWA) time-invariant dynamic system of the form (1), where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$, and $u \in \mathbb{R}^m$. We take into account constraints on the state and the input assuming that the state+input admissible set $\mathcal{X} \subseteq \mathbb{R}^{n+m}$ is a

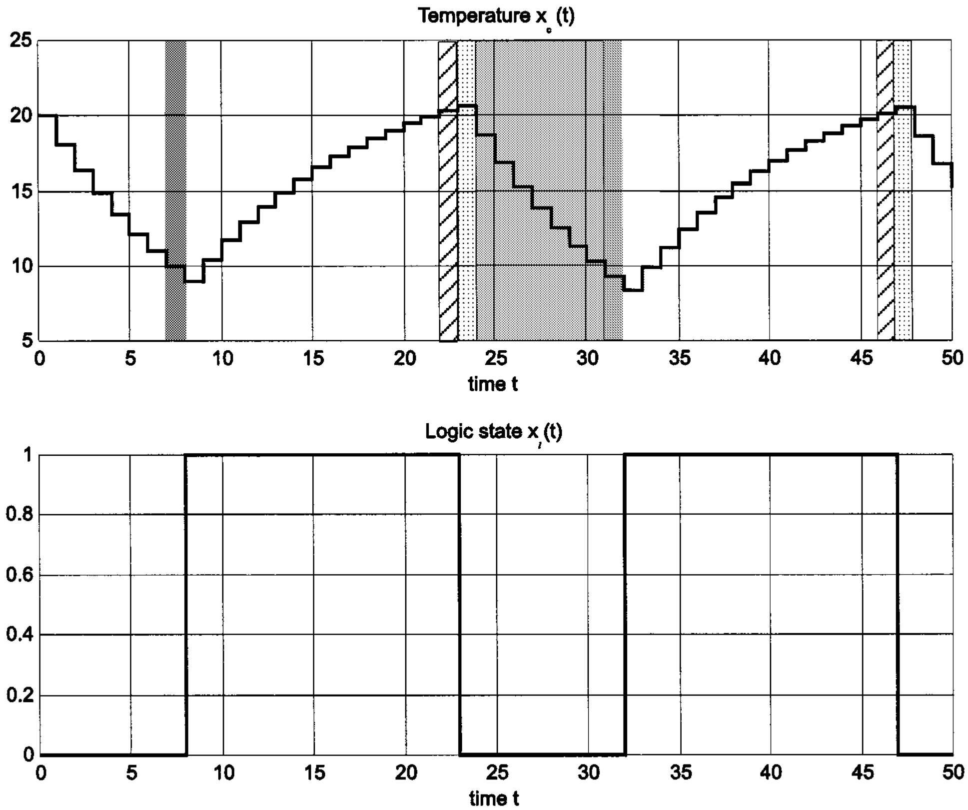

Fig. 2. Simulation of the temperature control system. The different regions where PWA component subsystems are active are depicted with different textures.

TABLE II
VALID COMBINATIONS $[\delta_1\ \delta_2\ \delta_3\ \delta_4]$ AND RESPECTIVE FUNCTIONS $z = G(x,u)$

$[\delta_1\ \delta_2\ \delta_3\ \delta_4]$	$z = G(x,u)$
[0 0 0 0] [0 1 1 0] [1 0 0 0]	$z = [0\ 0]x + [0]u + 0$
[0 0 1 1] [0 1 1 1] [1 0 0 1]	$z = [0\ 0]x + [1]u + 0$

convex and bounded polyhedron. Moreover, we suppose that $\mathcal{X}_i$, $i = 1, 2, \cdots, s$ forms a polyhedral partition[3] of $\mathcal{X}$.

A frequent representation of (1) arises in gain scheduling, where the linear model (and, consequently, the controller) is switched among a finite set of models, according to changes of the operating conditions.

PWA systems can be represented in the MLD form (2). The translation consists of defining logical δ_i variables $[\delta_i = 1] \leftrightarrow \left[\left[\begin{smallmatrix} x \\ u \end{smallmatrix}\right] \in \mathcal{X}_i\right]$ and imposing the exclusive-or condition $\bigoplus_{i=1}^{s} [\delta_i = 1]$. For details, the reader is referred to [4].

Conversely, we will show in Proposition 1 that every MLD model (2) is equivalent to a PWA system.

Before stating this general conversion result, we consider again the temperature control system of Section II-A. It is easy to check from (6a)–(9) that only the combinations $[\delta_1, \delta_2, \delta_3, \delta_4]$ reported in Table II are allowed. The corresponding relations between z and x, u are also reported in Table II.

[3]Each set $\mathcal{X}_i$ is a (not necessarily closed) convex polyhedron s.t. $\mathcal{X}_i \bigcap \mathcal{X}_j = \emptyset, \forall\, i \neq j, \bigcup_{i=1}^{s} \mathcal{X}_i = \mathcal{X}$.

As the switching is governed by changes of vector $\delta(t)$, it is intuitive that the number of regions in which the state space $\mathbb{R}^2$ is partitioned coincides with the number of valid δ combinations (i.e., six). To see this, consider, for example, $\delta = [0\ 0\ 0\ 0]'$. This gives $z = 0$, and, by substituting in (11b), the corresponding region is defined by the inequalities

$$\begin{bmatrix} -1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & -1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 1 \end{bmatrix} u \leq \begin{bmatrix} -m \\ M \\ 0 \\ 0 \\ -m_u \\ M_u \end{bmatrix} \tag{12}$$

where redundant constraints have been eliminated by using standard procedures based on linear programming. Moreover, from (11a) and $z = 0$, it follows that, in the region defined by (12), the state-update equations are

$$x(t+1) = \begin{bmatrix} \lambda & 0 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{13}$$

In Fig. 2, the different regions where PWA component subsystems are active are depicted by different textures.

Proposition 1: Consider generic trajectories $x(t)$, $u(t)$, $y(t)$ of an MLD system (2). Then there exist a polyhedral partition $\{\mathcal{X}_i\}_{i=1}^{s}$ of the state+input set

$$\mathcal{X} = \{(x,u) \in \mathbb{R}^{n_c} \times \{0,1\}^{n_\ell} \times \mathbb{R}^{m_c} \times \{0,1\}^{m_\ell} \\ \text{s.t. (2c) holds for some } \delta \in \{0,1\}^{r_\ell},\ z \in \mathbb{R}^{r_c}\}$$

and 5-tuples $(A_i, B_i, C_i, f_i, g_i)$, $i = 1, \cdots, s$, such that $x(t)$, $u(t)$, $y(t)$ satisfy (1).

Proof: In order to simplify the proof, without loss of generality, we assume that the logical components $x_{\ell i}$ of x_ℓ are also auxiliary variables, i.e., $\forall i = 1, \cdots, n_\ell\ \exists j$ such that $x_{\ell i} = \delta_j$. This is not a restrictive assumption, as typically the state transition of logical states derives from a logic predicate involving literals associated with components of $\delta(t)$ and $x_\ell(t)$, and the latter can be expressed again as additional auxiliary variables by simply adding the constraints $\delta_j(t) \leq x_{\ell i}(t), -\delta_j(t) \leq -x_{\ell i}(t)$ in (2c).

By the well posedness of system (2), given $x(t)$, $u(t)$, the vector $\delta(t)$ is uniquely defined, namely, $\delta(t) = F(x(t), u(t))$. Moreover, it only takes a value δ_i within a set of (at most) 2^{r_ℓ} values (corresponding to all possible 0–1 combinations). Let s be the number of valid combinations, i.e., the number of all different vectors $\delta \in \{0,1\}^{r_\ell}$ satisfying constraints (2c) for some $x(t)$, $u(t)$, $z(t)$. The idea is to partition the state+input space by grouping in regions $\mathcal{X}_i$ all $\left[\begin{smallmatrix} x(t) \\ u(t) \end{smallmatrix}\right]$ corresponding to the same binary vector $\delta_i = F(x(t), u(t))$. Let us fix $\delta(t) \equiv \delta_i$. The inequalities (2c) define a polyhedron $\mathcal{P}$ in $\mathbb{R}^{n+m+r_c}$. By the well posedness of $z(t)$, given a pair $x(t)$, $u(t)$, there exists only one value $z(t) \in \mathbb{R}^{r_c}$ satisfying (2c), namely, $z(t) = G(x(t), u(t))$. As all of the inequalities (2c) are linear, G is an affine function, namely,

$$\begin{aligned} z(t) = &K_{4i}x(t) + K_{1i}u(t) + K_{5i}, \\ &\forall x(t), u(t)\colon F(x(t), u(t)) = \delta_i \end{aligned} \tag{14}$$

and $\mathcal{P} \subset \mathbb{R}^{n+m+r_c}$ is a polyhedral set of dimension less than or equal to $n+m$ (for instance, if $n=1, m=0, r_c=1, \mathcal{P}$ would be a segment in $\mathbb{R}^2$). By substituting (14) in (2a) and (2b), we obtain

$$\begin{aligned} x(t+1) &= (A + B_3K_{4i})x(t) + (B_1 + B_3K_{1i})u(t) \\ &\quad + (B_2\delta_i + B_3K_{5i}) \\ y(t) &= (C + D_3K_{4i})x(t) + (D_1 + D_3K_{4i})u(t) \\ &\quad + (D_3K_{5i} + D_2\delta_i) \end{aligned}$$

which, by suitable choice of A_i, B_i, C_i, f_i, g_i, $i = 1, \cdots, s$, corresponds to (1) for

$$\mathcal{X}_i = \left\{ \begin{bmatrix} x \\ u \end{bmatrix}\colon \quad (E_3K_{4i} - E_4)x + (E_3K_{1i} - E_1)u \right. \\ \left. \leq (E_5 - E_3K_{5i} - E_2\delta_i) \right\}. \qquad \blacksquare$$

Remark 1: We stress the fact that the proof is based on a constructive argument. In fact, as was done in the temperature control system example, information on the description of the system can be used to derive (14), either from direct insight or automatically from the inequalities (2c).

Remark 2: From a computational point of view, both forms (1) and (2) have advantages. As in the case of linear time-varying systems, the former allows expressing the evolution of the system in a very compact way, for instance, when dealing with reach-set computation [6] (i.e., the computation of the set of states which are reachable from a given set of initial conditions). On the other hand, the latter allows inference, e.g., in a switching detection problem, namely, the problem of determining all possible new regions $\mathcal{X}_i$'s entered by a set of state vectors at the next time step. While the PWA form would be required for enumerating and checking for the nonemptiness of the intersections of the updated set with all of the regions $\mathcal{X}_i$, $i = 1, \cdots, s$, the MLD form instead can be conveniently exploited to solve the problem through mixed-integer linear optimization involving $\delta(t)$, $z(t)$ as free variables [6]. This indirectly moves the inference problem to the branch-and-bound strategy of the MILP solver.

IV. Observability

In this section, we consider observability of MLD systems (2) or, equivalently, PWA systems in view of Proposition 1.

Denote by $y(t, x, u)$ the output evolution at time t starting from the initial condition $x(0) = x$ and driven by the input $u(t)$, $t = 0, 1, \cdots$. We extend the definition of observability given in [22] and [29] to nonautonomous hybrid systems of the form (2).

Definition 1: Let $\mathcal{X}(0) \subseteq \mathbb{R}^{n_c} \times \{0,1\}^{n_\ell}$ be a set of initial states, and let $\mathcal{U} \subseteq \mathbb{R}^{m_c} \times \{0,1\}^{m_\ell}$ be a set of inputs. The MLD system (2) is *incrementally observable in T steps on $\mathcal{X}(0)$ uniformly with respect to $\mathcal{U}$* or simply *incrementally observable* if there exist two norms $\|\cdot\|_a$ (on $\mathbb{R}^{n_c+n_\ell}$) and $\|\cdot\|_b$ (on $\mathbb{R}^{p_c+p_\ell}$) and a positive scalar w such that $\forall x_1, x_2 \in \mathcal{X}(0)$ and $\forall$ input sequences $\{u(t)\}_{t=0}^{T-1} \subseteq \mathcal{U}$:

$$\sum_{t=0}^{T-1} \|y(t, x_1, u) - y(t, x_2, u)\|_b \geq w\|x_1 - x_2\|_a. \tag{15}$$

Remark 3: When including the input u in the definition of observability of nonlinear systems, some authors prefer asking that "$\exists$" (an input sequence $\{u(t)\}_{t=0}^{T-1} \subseteq \mathcal{U}$ such that $\cdots$) instead of "$\forall$." As typically an observer is used together with a controller, we have opted for the latter. In fact, in this situation, the output of the controller is not a sequence which is known *a priori*, and therefore observability should be required with respect to *all* possible input commands generated by the controller. Moreover, the class $\mathcal{U}$ of such commands is usually specified by the control system design, for instance, directly by limits on actuators. ■

Remark 4: The parameters T and w appearing in Definition 1 admit a practical interpretation. The scalar w can be viewed as an observability measure[4] for an incrementally observable system. For fixed initial states x_1 and x_2, the larger w, the more different the trajectories $y(t, x_1, u)$, $y(t, x_2, u)$ [from now on, we will write in short $y_1(t)$, $y_2(t)$]. Hence, in practice, one would fix a minimum observability level $w_{\min}$ and require that $w \geq w_{\min}$. If this condition is not fulfilled, we classify the system as *practically unobservable*. Practical unobservability also arises if Definition 1 is satisfied only for large T. Therefore, it is sensible to fix an upper bound $T_{\max}$ on T, and define an MLD system as practically observable when it satisfies Definition 1 with $T < T_{\max}$. ■

Condition (15) is simply an *incremental distinguishability* condition, i.e., it states that different initial states always give

[4]More precisely, one should use $\tilde{w} = \sup\{w > 0$ s.t. (15) holds$\}$ as the observability measure.

different outputs, independently of the applied input. However, although $y_1(t) \not\equiv y_2(t)$, in principle, there might be a component of x which is not observable. But this cannot be true. In fact, in this case, one could take two initial states such that the observable component is the same, which implies $y_1(t) \equiv y_2(t)$, $\forall t \geq 0$, thus violating Definition 1. In conclusion, the notions of incremental distinguishability and incremental observability coincide.

For bounded sets $\mathcal{X}(0)$, it is easy to verify that the term $w\|x_1 - x_2\|_a$ in Definition 1 could be substituted by a more general K_∞ function $\mathcal{W}(\|x_1 - x_2\|)$ (see [22] for the definition of the K_∞ class) such that $\mathcal{W}$ is lower and upper Lipschitz, i.e., there exist positive constants L_1, L_2 such that $L_1\|x\| \leq \mathcal{W}(\|x\|) \leq L_2\|x\|$. Therefore, we can conclude that Definition 1 is not much more restrictive than the O property given in [22].

A. Observability Counterexamples for PWA Systems

Definition 1 was formulated for the general class of hybrid systems described by the MLD form (2) or, equivalently, the PWA form (1). One might expect to exploit the structure of PWA systems to derive results about observability similar to those holding for linear systems. Below we show some counterexamples which undermine these hopes, even in the simpler case of autonomous PWL systems.

We first show that, in general, for PWL systems, the time of observability T has no relation to the order n of each subsystem, and therefore, if a PWL system is incrementally observable, nothing can be said, in general, about the minimum T such that Definition 1 holds.

Then, we show examples where the observability properties of a PWL system cannot be directly inferred from the observability properties of its linear subsystems. In fact, we will show that unobservable subsystems can be composed to build an observable PWL system, and vice versa, that the composition of observable subsystems can become unobservable.

1) A PWL System Incrementally Observable with T Arbitrarily Large: Consider the following system:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t+1) = \begin{cases} \begin{bmatrix} 1.1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t), & \text{if } \epsilon \leq x_1(t) < 1, \\ \begin{bmatrix} 0 & 0.9 \\ 0.9 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t), & \text{otherwise} \end{cases}$$
$$y(t) = x_1(t) \tag{16}$$

where $\epsilon > 0$ is fixed, and set

$$\mathcal{X}(0) = \left\{ \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} \in \mathbb{R}^2: \quad \epsilon < x_1(0) < 1 \right\}. \tag{17}$$

Then $y(t) = 1.1^t x_1(0)$, $\forall t \leq \overline{T}$ where

$$\overline{T} \triangleq \left\lceil \frac{\log \frac{1}{\epsilon}}{\log 1.1} \right\rceil$$

and $\lceil \cdot \rceil$ denotes the least upper integer. Moreover, $y(\overline{T}+1) = 0.9x_2(0)$, and therefore two initial states $x_1 = \left[\begin{smallmatrix} x_{11} \\ x_{12} \end{smallmatrix}\right]$, $x_2 = \left[\begin{smallmatrix} x_{11} \\ x_{22} \end{smallmatrix}\right]$, with $x_{12} \neq x_{22}$, are indistinguishable for $T \leq \overline{T}$. By Definition 1, system (16) is incrementally observable in $\overline{T}+2$ steps. In Fig. 3, we report the plot of the function $J(t) \triangleq \sum_{i=0}^{t} |y_1(t) - y_2(t)| - w\|x_1 - x_2\|_\infty$, and Definition 1 can be verified by visual inspection. We can render $\overline{T}$ arbitrarily large by choosing smaller and smaller values of ϵ (intuitively, the smaller the initial condition $x_1(0)$, the longer the time required for the output to overpass 1 and switch dynamics). By setting $\epsilon = 0$ in (16) and (17), it follows that the system (16) becomes incrementally observable on $\mathcal{X}(0)$ only in infinite steps, in the sense that, for each $\tilde{T}$, there exist initial states in $\mathcal{X}(0)$ that can be observed only after $T > \tilde{T}$ steps.

2) An Incrementally Observable PWL System whose Components are Unobservable: Consider the system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t+1) = \begin{cases} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t), & \text{if } x_1(t) > x_2(t) \\ \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t), & \text{if } x_1(t) \leq x_2(t) \end{cases} \tag{18a}$$

$$y(t) = \begin{cases} x_1(t), & \text{if } x_1(t) > x_2(t) \\ x_2(t), & \text{if } x_1(t) \leq x_2(t) \end{cases} \tag{18b}$$

whose component subsystems are unobservable. The evolutions of the state-space trajectories are depicted in Fig. 4.

Let $\mathcal{X}(0) \subset$ sector 1 $\cup$ sector 2 depicted in Fig. 4 be a bounded set of admissible initial states. If $x(0)$ lies in sector 1, we have $y(0) = x_1(0)$, and the first component of the initial state is immediately observed. However, since

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^t \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} x_1(0) \\ tx_1(0) + x_2(0) \end{bmatrix}$$

and $\mathcal{X}(0)$ is bounded, there exists a finite time $\overline{T} \geq 1$ such that the state enters sector 2. Then, $y(\overline{T}) = \overline{T}x_1(0) + x_2(0)$, and the second component $x_2(0)$ can be determined as well from the output knowledge. *Mutatis mutandis*, the same rationale applies when the initial state lies in sector 2. Then the system is incrementally observable in $\overline{T}$ steps on $\mathcal{X}(0)$. Note, however, that the system is not incrementally observable on initial sets $\mathcal{X}(0)$ intersecting sectors 3 or 4. Consider, in fact, an initial state that lies in sector 3 (or 4). From Fig. 4, it is clear that the state trajectory never crosses the line $x_1 = x_2$. Therefore, the evolutions will be governed by the first (the second) component of (18), thus implying the unobservability of the first (second) coordinate of the initial state.

3) An Unobservable PWL System whose Components are Observable: Consider the system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t+1) = \begin{cases} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t), & \text{if } x_1(t) > x_2(t) \\ \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t), & \text{if } x_1(t) \leq x_2(t) \end{cases}$$

$$y(t) = \begin{cases} x_1(t), & \text{if } x_1(t) > x_2(t) \\ x_2(t), & \text{if } x_1(t) \leq x_2(t) \end{cases}$$

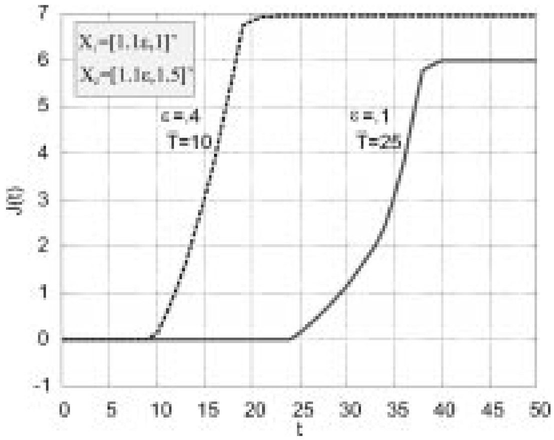

Fig. 3. States x_1, x_2 are indistinguishable for $\overline{T}$ steps.

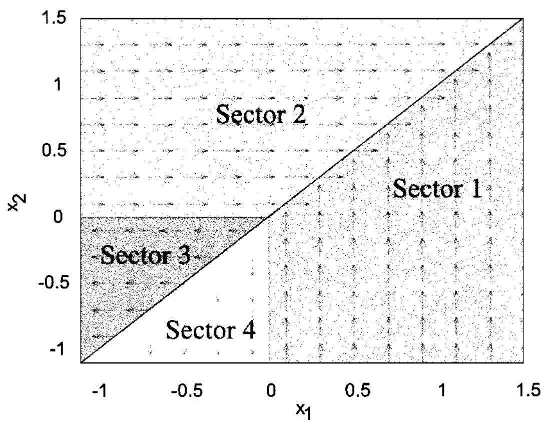

Fig. 4. State-space plane: $x(t+1) - x(t)$ normalized vector field.

whose components are observable. We partition again the state space as in Fig. 4. If the initial state lies in sector 3, by direct calculation, one has $y(0) = x_2(0)$ and $y(t) = 0$, $\forall t > 0$. Indeed, the state evolution for $t > 0$ is

$$x(t) = \begin{cases} [x_1(0) \quad 0]', & \text{if } t \text{ even} \\ [0 \quad x_1(0)]', & \text{if } t \text{ odd} \end{cases}$$

and $x_1(0) < 0$. Since the same rationale can be applied for initial states lying in sector 4, it can be concluded that the system is not incrementally observable on $\mathcal{X}(0) =$ sector 3 $\cup$ sector 4 (although it is easy to verify that the system is still incrementally observable on $\mathcal{X}(0) =$ sector 1 $\cup$ sector 2).

B. An Observability Test for Hybrid Systems

The purpose of this section is to derive an observability test for hybrid systems in the MLD form (2). In fact, the observability condition formulated in Definition 1 can be difficult to check, and thus one needs computationally tractable tests. Before stating Theorem 1, where we show that for MLD systems the incremental observability in T steps on $\mathcal{X}(0)$ and $\mathcal{U}$ is reduced to the solution of a mixed-integer linear program (MILP), we need some preliminary results.

Proposition 2: The MLD system (2) is incrementally observable if and only if there exists a scalar $w > 0$ such that

$$\min_{\substack{x_1 \in \mathcal{X}(0),\, x_2 \in \mathcal{X}(0) \\ u(t) \in \mathcal{U}, \\ t=0,\cdots,T-1}} \sum_{t=0}^{T-1} \|y_1(t) - y_2(t)\|_\infty - w\|x_1 - x_2\|_1 \geq 0. \tag{19}$$

Proof: The proof easily follows from the fact that all of the norms in finite-dimensional Euclidean spaces are equivalent. ∎

Proposition 2 proves the decidability of practical incremental observability (i.e., $w = w_{\min}$, as pointed out in Remark 4) over a finite time T. Unfortunately, the minimization problem (19) is, in general, nonconvex. In any case, the use of the norms $\|\cdot\|_\infty$ and $\|\cdot\|_1$ allows us to formulate it as an MILP problem. For this purpose, we need a technical lemma. In the sequel, $[x]_i$ will denote the ith element of vector x.

Lemma 1: Let $\mathcal{X}(0)$ be bounded. For two vectors x_1, x_2 in $\mathcal{X}(0)$, it holds that

$$\|x_1 - x_2\|_1 = \sum_{i=1}^{n} [x_1 - x_2]_i - 2[s]_i \tag{20a}$$
$$x_1 - x_2 \leq (M-m)'(\mathbf{1}_n - \mu) \tag{20b}$$
$$x_1 - x_2 \geq \sigma \mathbf{1}_n + (m - M - \sigma \mathbf{1}_n)'\mu \tag{20c}$$
$$s \leq (M-m)'\mu \tag{20d}$$
$$s \geq (m-M)'\mu \tag{20e}$$
$$s \leq x_1 - x_2 - (m-M)'(\mathbf{1}_n - \mu) \tag{20f}$$
$$s \geq x_1 - x_2 - (M-m)'(\mathbf{1}_n - \mu) \tag{20g}$$

where $\mu \in \{0,1\}^n$, $[M]_i = \max_{x \in \mathcal{X}(0)} x_i$, $[m]_i = \min_{x \in \mathcal{X}(0)} x_i$, $i = 1, \cdots, n$ and σ is a small tolerance (e.g., the machine precision).

Proof: By recalling Table I, (20b) and (20c) are obtained from the logical relation $[[\mu]_i = 1] \leftrightarrow [[x_1 - x_2]_i \leq 0]$, while (20d)–(20g) are obtained by translating the product $[s]_i = [x_1 - x_2]_i \mu_i$. Hence, $|[x_1 - x_2]_i| = [x_1 - x_2]_i - 2[s]_i$. ∎

Theorem 1: Let $\mathcal{X}(0)$ be bounded, and consider the following optimization problem:

$$J^* = \min_{\substack{\{\epsilon_t\}_{t=0}^{T-1},\, \epsilon_t \in \mathbb{R} \\ x_1 \in \mathcal{X}(0),\, x_2 \in \mathcal{X}(0) \\ s \in \mathbb{R}^n,\, \mu \in \{0,1\}^n, \\ \{\delta(t)\}_{t=0}^{T-1},\, \delta(t) \in \{0,1\}^{r_\ell} \\ \{z(t)\}_{t=0}^{T-1},\, z(t) \in \mathbb{R}^{r_c} \\ \{u(t)\}_{t=0}^{T-1} \subseteq \mathcal{U}}} \left\{ \sum_{t=0}^{T-1} \epsilon_t - w \cdot \left(\sum_{i=1}^{n} [x_1 - x_2]_i - 2[s]_i \right) \right\} \tag{21a}$$

subject to (2), (20b)–(20g), and

$$\mathbf{1}_p \epsilon_t \geq y_1(t) - y_2(t), \qquad t = 0, \cdots, T-1 \tag{21b}$$
$$\mathbf{1}_p \epsilon_t \geq y_2(t) - y_1(t), \qquad t = 0, \cdots, T-1. \tag{21c}$$

Then the MLD system (2) is incrementally observable in T steps on $\mathcal{X}(0)$ and $\mathcal{U}$ if and only if, for some $w > 0$, it holds that $J^* \geq 0$.

Proof: We start by proving necessity. Inequalities (21b) and (21c) imply that

$$\epsilon_t \geq \max_{i=1,\cdots,p} |[y_1(t)-y_2(t)]_i| = \|y_1(t)-y_2(t)\|_\infty. \quad (22)$$

By Lemma 1,

$$\|x_1-x_2\|_1 = \sum_{i=1}^{n} [x_1-x_2]_i - 2[s]_i. \quad (23)$$

Then, combining (22) and (23),

$$J^* \geq \min_{\substack{x_1\in\mathcal{X}(0),\, x_2\in\mathcal{X}(0),\\ u(t)\in\mathcal{U};\\ t=0,\cdots,T-1}} \sum_{t=0}^{T-1} \|y_1(t)-y_2(t)\|_\infty - w\|x_1-x_2\|_1. \quad (24)$$

In view of Proposition 2, the condition $J^*\geq 0$ follows from the incremental observability of system (2).

To show sufficiency, assume $J^*\geq 0$ and consider

$$J_1^* = \min_{\substack{x_1\in\mathcal{X}(0),\, x_2\in\mathcal{X}(0)\\ \{\delta(t)\}_{t=0}^{T-1},\, \delta(t)\in\{0,1\}^{r_\ell}\\ \{z(t)\}_{t=0}^{T-1},\, z(t)\in\mathbb{R}^{r_c}\\ \{u(t)\}_{t=0}^{T-1}\subseteq\mathcal{U}}} \sum_{t=0}^{T-1} \|y_1(t)-y_2(t)\|_\infty - w\|x_1-x_2\|_1 \quad (25)$$

subject to constraints (2), and let x_1^*, x_2^* denote the initial states that minimize (25). The variables $\{\epsilon_t\}_{t=0}^{T-1}$, μ, and s defined as

$$\epsilon_t \triangleq \|y(t,x_1^*)-y(t,x_2^*)\|_\infty$$
$$[s]_i \triangleq |[x_1^*-x_2^*]_i|, \qquad i=1,\cdots,n$$
$$[\mu]_i \triangleq \begin{cases} 1, & \text{if } [x_1^*-x_2^*]_i \leq 0\\ 0, & \text{if } [x_1^*-x_2^*]_i > 0 \end{cases} \qquad i=1,\cdots,n$$

are feasible for problem (21a). Thus, by optimality, $J_1^*\geq J^*\geq 0$, which proves incremental observability. ■

Theorem 1 is also helpful for designing an algorithm that checks the practical observability of an MLD system (see Remark 4. The procedure is summarized in the following steps.

Algorithm 1:
1) Choose $w_{\min}$ and $T_{\max}$ (see Remark 4).
2) Set $T=1$ and $w=w_{\min}$.
3) Solve the MILP (21a).
4) If $J^*\geq 0$, stop: the system is (practically) observable.
5) If $J^*<0$, increase T.
6) If $T>T_{\max}$, stop: the system is practically unobservable.
7) Go to step 3).

Remark 5: When the sets $\mathcal{X}(0)$, $\mathcal{U}$ are polytopes, the optimization problem (21) becomes an MILP in $T(1+r_c+m_c)+3n$ continuous variables and $T(r_\ell+m_\ell)+n$ integer variables. It is well known that, with the exception of particular structures, MILP's involving 0–1 variables are $\mathcal{NP}$ complete, which means that, in the worst case, the solution time grows exponentially with the number of integer variables [28]. Despite this combinatorial nature, several algorithmic approaches have been proposed and applied successfully to medium- and large-size application problems [17], and *branch-and-bound* methods were shown to be extremely successful.

In case the observability horizon T becomes large, solving such an optimization can become computationally intractable. As noted in the Introduction, this has to be expected because of the $\mathcal{NP}$-complete nature of the observability problem itself over a finite horizon [33]. Consider, for instance, the autonomous case (no input). By looking more closely at the MILP (21a), the main reason for the complexity is the presence of integer variables $\delta(t)$. Indeed, determining the optimal sequence $\delta(0),\cdots,\delta(t)$ corresponds to finding the sequence of the switching of linear dynamics, which leads to the worst case for observability.

Nevertheless, we now propose an algorithm which, although still exponential in the worst case, is very efficient on average. In fact, the difficulty caused by the (possibly huge) number of combinations of binary variables ($2^{T(r_\ell+m_\ell)+n}$, as pointed out in Remark 5) will be avoided in general by exploiting the equivalent PWA structure of hybrid systems.

C. An Efficient Observability Test for Hybrid Systems

We describe here a procedure to check the observability of PWA systems that reduces the computational complexity of Algorithm 1. We adopt tools developed for *formal verification* of hybrid systems [5], [6] where, basically, a set-reachability problem is solved through the exploration of all possible evolutions of the hybrid system from the set of initial states $\mathcal{X}(0)$.

The main advantage of adopting verification schemes is that they can exploit the PWA dynamics (1) when exploring the temporal evolution from the initial set $\mathcal{X}(0)$. More specifically, in the MILP problem (21a), the task of deciding in which order to explore the possible combinations of integer vectors $[\delta_0,\cdots,\delta_{T-1},\mu]'$ is assigned to the numerical solver. Clearly, most of the combinations will not be compatible with the constraints (2c). Next Algorithm 2 avoids considering these inadmissible combinations.

Let $\mathcal{R}$ be a list of subsets $\mathcal{R}_i\subseteq\mathbb{R}^n$, i.e., $\mathcal{R}=\{\mathcal{R}_1,\mathcal{R}_2,\cdots\}$, and let $\#\mathcal{R}$ denote its length (by convention, $\#\mathcal{R}=0$ iff $\mathcal{R}=\emptyset$). When a new set $\mathcal{R}_i$ is added or removed to the list, we write, respectively, $\mathcal{R}_i\rightarrow\mathcal{R}$ or $\mathcal{R}\rightarrow\mathcal{R}_i$. Finally, $\phi(\cdot,x_0)$ denote the state trajectory $x(\cdot)$ generated from the initial condition x_0. For the sake of simplicity, we consider a formulation for autonomous PWA systems, although the presence of inputs can be taken into account by adapting the verification algorithm proposed in [6].

Algorithm 2:
1) Set $T=1$ and $w=w_{\min}$.
2) For $i=1,\cdots,s$: if $\mathcal{X}(0)\cap\mathcal{X}_i\neq\emptyset$, then $\mathcal{X}(0)\cap\mathcal{X}_i\rightarrow\mathcal{R}$.
3) While $\mathcal{R}\neq\emptyset$:
   3.1) for i, $j=1,\cdots,\#\mathcal{R}$: solve

$$m_{ij} = \min_{\substack{x_1\in\mathcal{X}(0),\, x_2\in\mathcal{X}(0)\\ \phi(T-1,x_1)\in\mathcal{R}_i,\, \phi(T-1,x_2)\in\mathcal{R}_j}} \cdot \sum_{t=0}^{T-1} \|y_1(t)-y_2(t)\|_\infty - w\|x_1-x_2\|_1$$

3.2) for $i=1,\cdots,\#\mathcal{R}$: if $m_{ij}\geq 0$, $\forall j=1,\cdots,\#\mathcal{R}$, then $\mathcal{R}\to\mathcal{R}_i$
3.3) if $\mathcal{R}=\emptyset$, STOP: the PWL system is practically incrementally observable in T steps
3.4) set $\overline{\mathcal{R}}=\emptyset$
3.5) for $i=1\cdots\#\mathcal{R}$
  3.5.1) let l be the index such that $\mathcal{R}_i\subset\mathcal{X}_l$
  3.5.2) for $j=1,\cdots,s$: if $(A_l\mathcal{R}_i+f_l)\cap\mathcal{X}_j\neq\emptyset$, $\{(A_l\mathcal{R}_i+f_l)\cap\mathcal{X}_j\}\to\overline{\mathcal{R}}$
3.6) increase T
3.7) if $T>T_{\max}$, STOP: the system is practically unobservable
3.8) $\mathcal{R}=\overline{\mathcal{R}}$.

Algorithm 2 computes the evolution from the initial set $\mathcal{X}(0)$ in order to explore all possible state trajectories $x(t)$. In any case, since after T steps there may exist some subsets of $\mathcal{X}(0)$ whose elements can be observed (i.e., distinguished from the other states in $\mathcal{X}(0)$ in T steps), the algorithm avoids further propagating such states. More precisely, at step 3.1, the set $\cup_{i=1}^{\#\mathcal{R}}\mathcal{R}_i$ collects the evolution of all of the initial states that are not observable in $T-1$ steps. Then, the algorithm checks if all of the initial states x_1 such that $\phi(T-1,x_1)\in\mathcal{R}_i$ are distinguishable from the initial states x_2 satisfying $\phi(T-1,x_2)\in\mathcal{R}_j$. This is done in step 3.1 by computing m_{ij} because, analogously to (19), the distinguishability condition corresponds to $m_{ij}\geq 0$. In particular, if $m_{ij}\geq 0, \forall j=1,\cdots,\#\mathcal{R}$, the subset of initial states evolving in $\mathcal{R}_i$ is observable (in T steps). Then, there is no need to further consider the evolution of the set $\mathcal{R}_i$, and it is removed from the list (step 3.2). It is also apparent that the practical incremental observability of the PWA system coincides with the condition $\mathcal{R}=\emptyset$ (step 3.3).

The one-step evolution of the sets $\mathcal{R}_i$ is performed in step 3.4. Note that, from steps 2 and 3.5.2, it follows that each set $\mathcal{R}_i$ belongs at most to a single region $\mathcal{X}_i$ of the state space. This ensures that the index l (step 3.5.1) is always well defined and, in step 3.5.2, every set $\mathcal{R}_i$ evolves according the state equation of the region which it belongs to. Moreover, if the set $(A_l\mathcal{R}_i+f_l)$ intersects k regions, it is split into k new sets (each one belonging only to a single region $\mathcal{X}_i$) that are then added to the updated list $\overline{\mathcal{R}}$.

Algorithm 2 is more suitable for implementation when the initial set $\mathcal{X}(0)$ is a bounded polyhedron. In this case, by means of the update step 3.5.2, every set $\mathcal{R}_i$ is a polytope as well. Moreover, the minimization in step 3.1 becomes a mixed-integer linear program. Actually, following the rationale of Theorem 1, it is easy to prove that

$$m_{ij}=\min_{\substack{x_1\in\mathcal{X}(0),\,x_2\in\mathcal{X}(0)\\ \phi(T-1,x_1)\in\mathcal{R}_i,\\ \phi(T-1,x_2)\in\mathcal{R}_j,\\ s\in\mathbb{R}^n,\,\mu\in\{0,1\}^n}}\left\{\sum_{t=0}^{T-1}\epsilon_t-w\cdot\left(\sum_{i=1}^{n}[x_1-x_2]_i-2[s]_i\right)\right\}\tag{26}$$

subject to (1), (20b)–(20g), (21b), and (21c). Note that each MILP problem (26) involves only n integer variables versus the $Tr_\ell+n$ required for problem (21a). This smaller number of integer variables is the main reason for the computational effectiveness of Algorithm 2.

Remark 6: In view of Proposition 1, it is apparent that the property of incremental observability does not change when switching between PWA and MLD representations. Therefore, Algorthm 2 is also suitable for checking the (practical) incremental observability of an autonomous MLD system.

As an illustrative example, we apply Algorthm 2 to system (16) with $\epsilon=0.4$ and $\mathcal{X}(0)=[0.4,0.99]\times[-2,2]$. The observability parameters are chosen as $w_{\min}=10^{-3}$ and $T_{\max}=20$. In Fig. 5, the sets in the list $\mathcal{R}$ are plotted in different gray levels at times $T=1$, 4, 8, 12. Note that, due to the evolution of the system and the fathoming criterion, the vertical set in the region $\epsilon\leq x_1(t)<1$ progressively shrinks and, for $T>12$, it disappears. Algorthm 2 terminates by finding practical incremental observability in 12 steps, in accordance with the analytic results of Section IV-A-1 (CPU time: 22.8 s on a Pentium II 300 running Matlab 5.3).

D. A Deadbeat Observer for Hybrid Systems

Proposition 3 provides a deadbeat observer for hybrid systems (2). In a certain sense, it is a counterpart of [30, Theorem 2.10].

Proposition 3: Let $\hat{x}(t-T+1),\cdots,\hat{x}(t)$ be the minimizing sequence of the following least squares problem:

$$\min_{\hat{x}(t-T+1),\{\hat{\delta}(t-k),\hat{z}(t-k)\}_{k=0}^{T-1}}\sum_{k=0}^{T-1}\|\hat{y}(t-k)-y(t-k)\|_2^2$$

subject to

$$\begin{aligned}\hat{x}(k+1)&=A\hat{x}(k)+B_1u(k)+B_2\hat{\delta}(k)+B_3\hat{z}(k)\\ \hat{y}(k)&=C\hat{x}(k)+D_1u(k)+D_2\hat{\delta}(k)+D_3\hat{z}(k)\\ E_2\hat{\delta}(k)+E_3\hat{z}(k)&\leq E_1u(k)+E_4\hat{x}(k)+E_5\\ k&=t-T+1,\cdots,t\end{aligned}\tag{27}$$

where $y(t-k)$, $u(t-k)$, $k=0,\cdots,T-1$ are the collection of past T inputs and outputs, and T is any time horizon such that Definition 1 is satisfied. Then $\hat{x}(t)$ is an estimate of the state $x(t)$ and $\hat{x}(t)\equiv x(t)$, $\forall\, t\geq T_{\min}$, where $T_{\min}\leq T$ is the minimum time horizon for observability.

Proof: After T input/output pairs have been collected, the minimum in problem (27) is 0, and the minimizer is $\hat{x}(t-T+1)=x(t-T+1)$ because, otherwise, there would exist a state $\hat{x}(t-T+1)$ which is indistinguishable from the true state $x(t-T+1)$ based on the observed output sequence. ■

Note that the optimization problem (27) is a mixed-integer quadratic program, for which efficient solvers exist [16]. An MILP formulation can be obtained by using 1- or ∞-norms, as in Theorem 1, instead of the squared 2-norm.

As observed in Remark 5, the optimization problem (27) is $\mathcal{NP}$ complete, and therefore computationally expensive for large T. Again, the complexity arises from the need for determining the sequence of switches of the linear dynamics which has occurred between time $t-T$ and time t. Nevertheless, the $\mathcal{NP}$ completeness of the problem of solving (27) does not imply that simpler observers do not exist for hybrid systems.

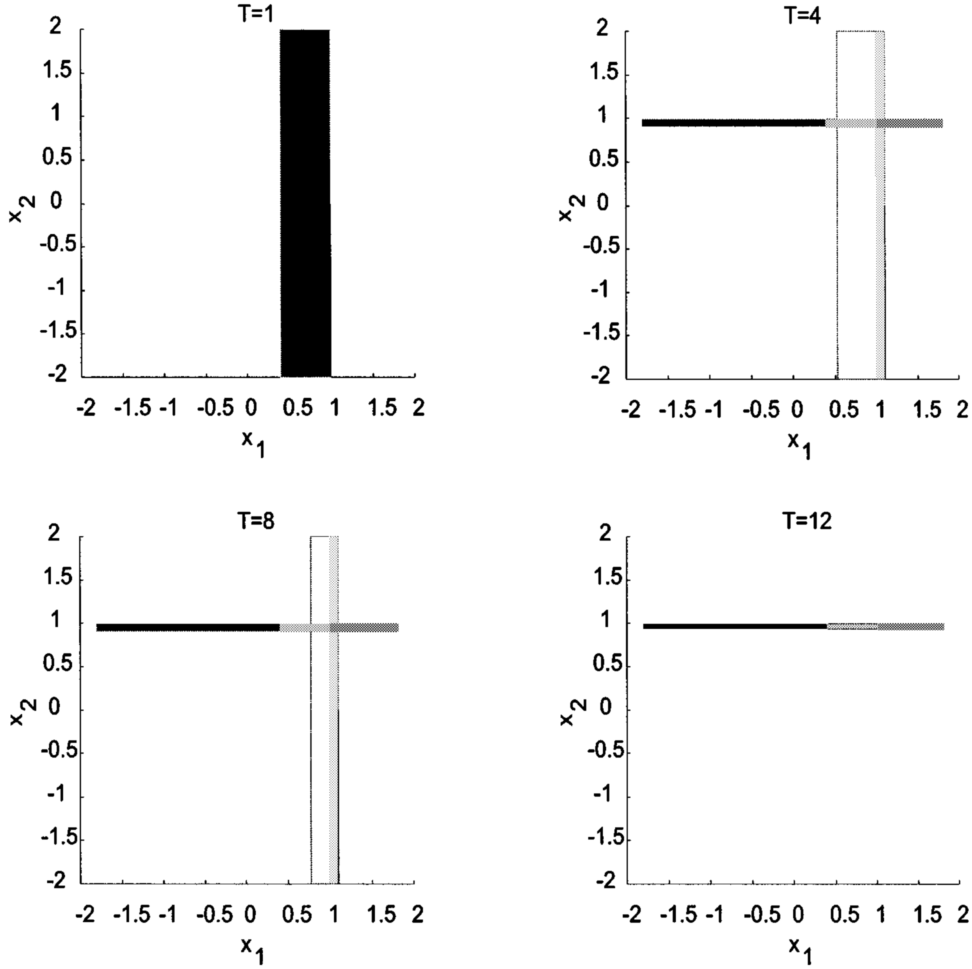

Fig. 5. Evolution of the list $\mathcal{R}$ of Algorithm 2: different sets $\mathcal{R}_i$ are depicted by different gray levels. Snapshots at $T = 1$, 4, 8, 12. After $T = 12$, all of the sets $\mathcal{R}_i$ disappear.

V. Controllability

We introduce the following definition of controllability for MLD systems.

Definition 2: Let $\mathcal{X}(0)$ and $\mathcal{X}_f$ be nonempty sets of initial and final states, respectively. The MLD system (2) is controllable in T steps from $\mathcal{X}(0)$ to $\mathcal{X}_f$ if, $\forall\, x_0 \in \mathcal{X}(0)$, there exists an admissible input sequence $\{u(t)\}_{t=0}^{T-1}$ yielding

$$x(T) \in \mathcal{X}_f. \tag{28}$$

If $\mathcal{X}(0)$ and $\mathcal{X}_f$ are singletons (i.e. $\mathcal{X}(0) = \{x_0\}$ and $\mathcal{X}_f = \{\overline{x}\}$), Definition 2 reduces to a classical controllability notion [34]. In this case, our definition of controllability is instrumental for checking if the state can be driven from a perfectly known initial condition to a desired state $\overline{x}$ (usually an equilibrium state) by using a suitable control sequence $\{u(t)\}_{t=0}^{T-1}$. In any case, letting $\mathcal{X}(0)$ be a general set, we also take into account the case of incompletely specified initial conditions. Moreover, in many situations, the control specifications demand driving a system into a set of *safe* states $\mathcal{X}_f$ [5]. It is apparent that Definition 2 also embraces this scenario.

Remark 7: In principle, one might be concerned about the practical meaning of Definition 2. More specifically, due to physical limitations, a user may be interested in controlling the plant using only a bounded input. We point out that, even if such a constraint does not appear explicitly in Definition 2, it can be easily included in inequalities (2c).

A. Controllability Counterexamples for PWA Systems

Analogously to the observability notion, we specialize the controllability definition to PWL systems. Again, through some counterexamples, we will show that this property cannot be inferred from the controllability of the component subsystems.

1) An Uncontrollable PWL System whose Components are Controllable: Consider the system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t+1) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t) + \begin{cases} \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), & \text{if } x_1(t) > x_2(t) \\ \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t), & \text{if } x_1(t) \le x_2(t) \end{cases} \tag{29}$$

whose components

$$\left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right), \quad \left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right)$$

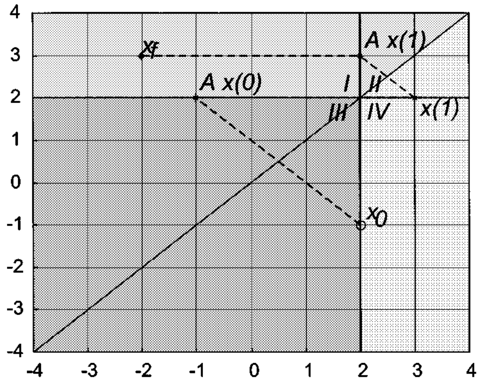

Fig. 6. State-space for system (29), whose components are completely controllable: Region III is not reachable from x_0.

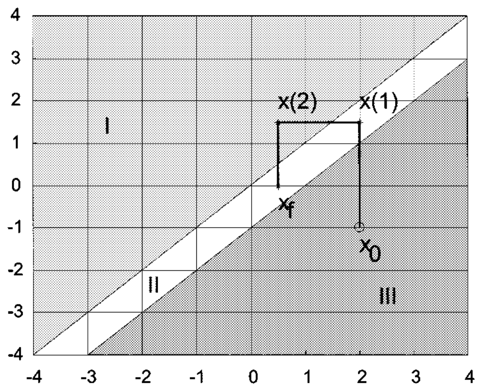

Fig. 7. State space for system (30), whose components are uncontrollable.

are completely controllable. Let $x_0 = \begin{bmatrix} x_{10} \\ x_{20} \end{bmatrix}$ be the initial state, and consider the partition of the state space depicted in Fig. 6. The sectors I–IV are obtained by intersecting the lines $x_1 = \max\{x_{10}, x_{20}\}$, $x_2 = \max\{x_{10}, x_{20}\}$. It is easy to verify that only the sectors I, II, IV, are completely reachable from x_0, while III is not reachable. For instance, the point $x_f = (-2, 3)$ can be reached from $x_0 = (2, -1)$ by applying the input $u(0) = 4, u(1) = -4$, but no input can steer x_0 to the origin. In general, the PWL system (29) is controllable to 0 from x_0 if and only if $0 \in \text{I} \bigcup \text{II} \bigcup \text{IV}$, where we point out that sectors I–IV depend on x_0. Therefore, x_0 is controllable to the origin if and only if $x_{01} < 0, x_{02} < 0$.

2) A Controllable PWL System whose Components are Uncontrollable: Consider the system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t+1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}(t) + \begin{cases} \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t), & \text{if } x_2(t) - 1 > x_1(t) > x_2(t) \\ \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), & \text{if } x_1(t) \le x_2(t) \end{cases}$$

whose components

$$\left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right), \quad \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right)$$

are uncontrollable. The three regions in which the state space is partitioned are depicted in Fig. 7. It is easy to verify by inspection that every initial state in $\mathcal{X}(0) = \mathbb{R}^n$ can be controlled to any other state in at most three steps. This is, for instance, the situation depicted in Fig. 7, where the point $x_f = (0.5, 0)$ is steered from $x_0 = (2, -1)$ by applying the input $u(0) = 2.5$, $u(1) = u(2) = -1.5$.

B. Controllability Tests for Hybrid Systems

In this section, we discuss numerical tests for checking the controllability of an MLD system. We first notice that Definition 2 can be translated into the following *mixed-integer feasibility test* (MIFT):

$$\begin{cases} x(0) \in \mathcal{X}(0) \\ x(T) \in \mathcal{X}_f \\ x(t+1) = Ax(t) + B_1 u(t) + B_2 \delta(t) + B_3 z(t) \\ E_2 \delta(t) + E_3 z(t) \le E_1 u(t) + E_4 x(t) + E_5 \\ t = 0, 1, \cdots, T. \end{cases} \quad (30)$$

The feasibility test (30) is called a *verification* problem in the hybrid system literature. Unfortunately, solving the MIFT for large T becomes prohibitive. In fact, each problem (30) is $\mathcal{NP}$ complete, which means that, in the worst case, the required computation time grows exponentially with T. Despite this strong theoretical limitation, a verification algorithm for the general class of MLD systems under the assumption that both $\mathcal{X}(0)$ and $\mathcal{X}_f$ are polyhedra was proposed [5], [6]. This procedure is based on a sequence of linear and mixed-integer linear programs, and can be adopted as a numerical controllability test. Various other verification techniques have been proposed in the literature [1], [2], [23].

VI. Conclusions

In this paper, we illustrated, through a number of counterexamples, the complexity of the observability and controllability properties of PWA and hybrid systems. After proving the equivalence between PWA and hybrid MLD systems, we exploited this equivalence to derive observability and controllability tests which are numerically appealing.

References

[1] R. Alur, C. Courcoubetis, T. A. Henzinger, and P.-H. Ho, "Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems," in *Hybrid Systems*, A. P. Ravn, R. L. Grossman, A. Nerode, and H. Rischel, Eds. Berlin: Springer-Verlag, 1993, vol. 736, Lecture Notes in Computer Science, pp. 209–229.
[2] A. Asarin, O. Maler, and A. Pnueli, "On the analysis of dynamical systems having piecewise-constant derivatives," *Theoret. Comput. Sci.*, vol. 138, pp. 35–65, 1995.
[3] A. Bemporad, D. Mignone, and M. Morari, "Fault detection and state estimation for hybrid systems," in *Proc. Amer. Contr. Conf.*, 1999, pp. 2471–2475.
[4] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, pp. 407–428, Mar. 1999.
[5] ——, "Verification of hybrid systems via mathematical programming," in *Hybrid Systems: Computation and Control*, F. W. Vaandrager and J. H. van Schuppen, Eds. Berlin: Springer-Verlag, 1999, vol. 1569, Lecture Notes in Computer Science, pp. 31–45.
[6] A. Bemporad, F. D. Torrisi, and M. Morari, "Optimization-based verification and stability characterization of piecewise affine and hybrid systems," in *Hybrid Systems: Computation and Control*, B. Krogh and N. Lynch, Eds. Berlin, Germany: Springer-Verlag, 2000, Lecture Notes in Computer Science, pp. 45–58.

[7] V. D. Blondel, O. Bournez, P. Koiran, and J. N. Tsitsiklis, "The stability of saturated linear systems is undecidable," in *Proc. Amer. Contr. Conf.*, 2000, p. 1684.

[8] V. D. Blondel and J. N. Tsitsiklis, "Complexity of stability and controllability of elementary hybrid systems," *Automatica*, vol. 35, pp. 479–489, Mar. 1999.

[9] M. Branicky, "Stability of switched and hybrid systems," in *Proc. Conf. Decision Contr.*, 1994, pp. 3498–3503.

[10] ——, "Multiple Lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Trans. Automat. Contr.*, vol. 43, pp. 475–482, Apr. 1998.

[11] M. S. Branicky, V. S. Borkar, and S. K. Mitter, "A unified framework for hybrid control: Model and optimal control theory," *IEEE Trans. Automat. Contr.*, vol. 43, pp. 31–45, 1998.

[12] T. M. Cavalier, P. M. Pardalos, and A. L. Soyster, "Modeling and integer programming techniques applied to propostional calculus," *Comput. Oper. Res.*, vol. 17, no. 6, pp. 561–570, 1990.

[13] D. Christiansen, *Electronics Engineers' Handbook*, 4th ed. New York: IEEE Press/McGraw-Hill, 1997.

[14] J. Ezzine and A. H. Haddad, "On the controllability and observability of hybrid systems," *Int. J. Contr.*, vol. 49, pp. 2045–2055, 1989.

[15] G. Ferrari-Trecate, D. Mignone, and M. Morari, "Moving horizon estimation for hybrid systems," in *Proc. American Control Conf.*, 2000, Tech. Rep. AUT99-21, pp. 1684–1688.

[16] R. Fletcher and S. Leyffer, "Numerical experience with lower bounds for MIQP branch-and-bound," *SIAM J. Optimiz.*, vol. 8, pp. 604–616, May 1998.

[17] C. A. Floudas, *Nonlinear and Mixed-Integer Optimization*. Oxford, England: Oxford Univ. Press, 1995.

[18] D. Goshen-Meskin and I. Y. Bar-Itzhack, "Observability analysis of piecewise constant systems—Part I: Theory," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 28, pp. 1056–1067, Oct. 1992.

[19] A. Hassibi and S. Boyd, "Quadratic stabilization and control of piecewise-linear systems," in *Proc. Amer. Contr. Conf.*, 1998, pp. 3659–3664.

[20] T. A. Henzinger, P. W. Kopke, A. Puri, and P. Varaiya, "What's decidable about hybrid automata?," in *Proc. 27th ACM Symp. Theory of Comput.*, 1995, pp. 373–381.

[21] M. Johansson and A. Rantzer, "Computation of piecewise quadratic Lyapunov functions for hybrid systems," *IEEE Trans. Automat. Contr.*, vol. 43, pp. 555–559, 1998.

[22] S. S. Keerthi and E. G. Gilbert, "Optimal infinite-horizon feedback control laws for a general class of constrained discrete-time systems: Stability and moving-horizon approximations," *J. Opt. Theory Appl.*, vol. 57, pp. 265–293, 1988.

[23] Y. Kesten, A. Pnueli, J. Sifakis, and S. Yovine, "Integration graphs: A class of decidable hybrid systems," in *Hybrid Systems*, R. L. Grossman, A. Nerode, A. P. Ravn, and H. Rischel, Eds. Berlin, Germany: Springer-Verlag, 1993, vol. 736, Lecture Notes in Computer Science, pp. 179–208.

[24] G. Labinaz, M. M. Bayoumi, and K. Rudie, "A survey of modeling and control of hybrid systems," *Ann. Rev. Contr.*, vol. 21, pp. 79–92, 1997.

[25] J. Lygeros, D. N. Godbole, and S. Sastry, "A game theoretic approach to hybrid system design," in *Hybrid Systems III*, R. Alur and T. Henzinger, Eds. Berlin: Springer-Verlag, 1996, vol. 1066, Lecture Notes in Computer Science, pp. 1–12.

[26] J. Lygeros, C. Tomlin, and S. Sastry, "Controllers for reachability specifications for hybrid systems," *Automatica*, vol. 35, no. 3, pp. 349–370, 1999.

[27] D. Mignone, A. Bemporad, and M. Morari, "A framework for control, fault detection, state estimation and verification of hybrid systems," in *Proc. Amer. Contr. Conf.*, 1999, pp. 134–138.

[28] R. Raman and I. E. Grossmann, "Relation between MILP modeling and logical inference for chemical process synthesis," *Comput. Chem. Eng.*, vol. 15, no. 2, pp. 73–84, 1991.

[29] C. V. Rao and J. B. Rawlings, "Nonlinear moving horizon state estimation," in *Proc. Int. Symp. Nonlinear Model Predictive Contr.: Assessment and Future Directions*, Ascona, Switzerland, 1998.

[30] E. D. Sontag, "Nonlinear regulation: The piecewise linear approach," *IEEE Trans. Automat. Contr.*, vol. AC–26, pp. 346–358, Apr. 1981.

[31] ——, "Real addition and the polynomial hierarchy," *Inform. Processing Lett.*, vol. 20, pp. 115–120, 1985.

[32] ——, "Controllability is harder to decide than accessibility," *SIAM J. Contr. Optimiz.*, vol. 26, pp. 1106–1118, 1988.

[33] ——, "Interconnected automata and linear systems: A theoretical framework in discrete-time," in *Hybrid Systems III—Verification and Control*, R. Alur, T. A. Henzinger, and E. D. Sontag, Eds. Berlin: Springer-Verlag, 1996, vol. 1066, Lecture Notes in Computer Science, pp. 436–448.

[34] ——, *Mathematical Control Theory*. New York: Springer-Verlag, 1998.

[35] M. L. Tyler and M. Morari, "Propositional logic in control and monitoring problems," *Automatica*, vol. 335, no. 4, pp. 565–582, 1999.

[36] V. I. Utkin, "Variable structure systems with sliding modes," *IEEE Trans. Automat. Contr.*, vol. AC-22, no. 2, pp. 212–222, Apr. 1977.

[37] M. A. Wicks and P. Peleties, "Construction of piecewise Lyapunov functions for stabilizing switched systems," in *Proc. Conf. Decision Contr.*, 1994, pp. 3492–3497.

[38] H. P. Williams, "Logical problems and integer programming," *Bull. Inst. Math. Appl.*, vol. 13, pp. 18–20, 1977.

[39] ——, *Model Building in Mathematical Programming*, 3rd ed. New York: Wiley, 1993.

Alberto Bemporad was born in Florence, Italy, in 1970. He received the M.S. degree in electrical engineering in 1993 and the Ph.D. degree in systems science in 1997, both from the University of Florence, Italy.

He spent the academic year 1996/1997 at the Center for Robotics and Automation, Department of Systems Science & Mathematics, Washington University, St. Louis, MO, as a Visiting Researcher. In 1997, he joined the Automatic Control Lab, ETH, Zurich, Switzerland, as a Postdoctoral Fellow, and in 1999, he was appointed Assistant Professor at the University of Siena, Italy.

Dr. Bemporad received the IEEE Centre and South Italy section "G. Barzilai" and the AEI (Italian Electrical Association) "R. Mariani" best graduate awards. He has published papers in the area of model predictive control, hybrid systems, and robotics.

Giancarlo Ferrari-Trecate was born in Casorate Primo, Italy, in 1970. He received the "Laurea" degree in computer engineering in 1995, and the Ph.D. degree in electronic and computer engineering in 1999, both from the University of Pavia.

In 1998, he was a Visiting Researcher at the Neural Computing Research Group, University of Birmingham, UK. In 1998, he joined ETH as a Postdoctoral Fellow. In 1999, he won the "Assegno di Ricerca" at the University of Pavia. His research interests include linear periodic systems, Bayesian learning, fuzzy systems, and hybrid systems.

Manfred Morari received the diploma from ETH Zurich, and the Ph.D. degree from the University of Minnesota.

He was appointed Head of the Automatic Control Laboratory at the Swiss Federal Institute of Technology (ETH) in Zurich in 1994. Before that, he was the McCollum–Corcoran Professor and Executive Officer for Control and Dynamical Systems at the California Institute of Technology. His interests are in hybrid systems and the control of biomedical systems.

In recognition of his research, Dr. Morari received numerous awards, among them the Eckman Award of the AACC, the Colburn Award and the Professional Progress Award of the AIchE, and he was elected to the National Academy of Engineering (U.S.). He has held appointments with Exxon R & E and ICI, and has consulted internationally for a number of major corporations.