Proceedings of the 45th IEEE Conference on Decision & Control
Manchester Grand Hyatt Hotel
San Diego, CA, USA, December 13-15, 2006

Control of Rectangular Multi-Affine Hybrid Systems

L.C.G.J.M. Habets, M. Kloetzer, and C. Belta

Abstract— **We study the problem of feedback control for a class of non-linear hybrid systems characterized by rectangular invariants and multi-affine dynamics, which we call Rectangular Multi-Affine Hybrid Systems. The goal is to find initial states and feedback control strategies so that all trajectories of the closed loop system satisfy arbitrary specifications given as temporal logic formulas over the set of discrete states of the system. Sufficient conditions for solvability are obtained in terms of sets of linear inequalities. If these conditions are satisfied, a control strategy is automatically constructed. The computation consists of polyhedral set operations, construction of Büchi automata from linear temporal logic formulas, and searches on graphs.**

I. Introduction

Temporal logics [6] were developed for specifying the correctness of digital circuits and computer programs. However, due to their resemblance to natural language, their expressivity, and the existence of off-the-shelf algorithms for model checking, temporal logics have the potential to impact several other areas of engineering. Analysis of systems with continuous dynamics based on qualitative simulations and temporal logic was proposed in [18], [7]. Control of linear systems from temporal logic specifications has been considered in both discrete [19] and continuous time [15]. The use of temporal logic for task specification and controller synthesis in robotics has been advocated in [8], [16]. In the area of systems biology, the qualitative behavior of genetic circuits can be expressed in temporal logic, and model checking can be used for analysis, as suggested in [2].

Hybrid systems combine continuous and discrete dynamics and have been found to be very useful in modelling processes from various areas including automated highway systems, embedded automotive and avionic controllers, manufacturing systems, real-time communication networks, cooperative robotics, and molecular networks. Such systems are characterized by a set of *continuous dynamics*, a set of predicates over the continuous state space giving *invariants* and *guards*, and a set of maps modelling *transitions* and *resets*. The discrete states can be seen as a set of symbols labelling the invariants. The semantics of such systems can be conceptually defined as follows: the continuous state of the system evolves along a given vector field as long as the corresponding invariant is true, and no guard is hit. When this happens, a transition occurs to another invariant (discrete state), possibly with an associated reset of the continuous state in the new invariant (see [1] for detailed definitions).

In this paper, we focus on Rectangular Multi-Affine Hybrid Systems (RMAHS), which have (multi-dimensional) rectangular invariants and multi-affine (*i.e.,* affine in each continuous state component) vector fields. This class of dynamics is rather large, and includes the celebrated Euler, Volterra and Lotka-Volterra equations, attitude and velocity control systems for aircraft and underwater vehicles [3], and models of genetic regulatory networks [4]. We assume the guards are given by the facets of the invariants and the resets are arbitrary. For such systems, we consider control specifications given as arbitrary LTL_{-X} formulas over the discrete states of the system. Intuitively, such specifications are logical and temporal statements about the reachability of the invariants by the continuous trajectories of the system. We derive sufficient conditions for the existence of feedback control strategies in terms of sets of linear inequalities. If these conditions are satisfied, a control strategy in the form of a multi-affine state feedback controller is automatically constructed. The computation consists of polyhedral set operations, construction of Büchi automata from linear temporal logic formulas, and searches on graphs.

This paper is related to [11], [12], [5]. The results on facet reachability for simplices by trajectories of affine vector fields derived in [11] are used to solve a very similar problem to the one we consider in this paper - the difference is that the invariants and guards of the hybrid system are simplices, the dynamics are affine, and the specifications are simpler, in the form of reach-avoid problems. The starting point for this paper is [5], where the problem of reachability of a facet of a rectangle by the trajectories of a continuous multi-affine system was studied. In order to deal with the more general problem formulated here, in this paper we present an extension of the main reachability theorem from [5].

Section II provides some preliminaries necessary throughout the paper. The problem is formulated in Section III. In Section IV we derive sufficient conditions that guarantee that all trajectories of a multi-affine system leave a rectangle through one or more specified facets. The control strategy providing a solution to the main problem is presented in Section V, and an example is presented in Section VI. We conclude with final remarks in Section VII.

II. Preliminaries

A. Transition systems and linear temporal logic

A *transition system* is a tuple $T = (Q, Q_0, \rightarrow, \Pi, \vDash)$, where Q is a set of states, $Q_0 \subseteq Q$ is a set of initial states,

This work was partially supported by NSF CAREER 0611926 and NSF 0611925 at Boston University

L.C.G.J.M. Habets is with the Department of Mathematics and Computer Science, Technische Universiteit Eindhoven, P.O. Box 513, NL-5600 MB Eindhoven, The Netherlands l.c.g.j.m.habets@tue.nl

M. Kloetzer and C. Belta are with the Center for Information and Systems Engineering, Boston University, 15 Saint Mary's St., Brookline, MA 02446, USA kmarius,cbelta@bu.edu

1-4244-0171-2/06/$20.00 ©2006 IEEE.

$\rightarrow \subseteq Q \times Q$ is a transition relation, Π is a finite set of atomic propositions, and $\vDash \subseteq Q \times \Pi$ is a satisfaction relation. For an arbitrary proposition $\pi \in \Pi$, we define $[[\pi]] = \{q \in Q | q \vDash \pi\}$ as the set of all states satisfying it. Conversely, for an arbitrary state $q \in Q$, let $\Pi_q = \{\pi \in \Pi \,|\, q \vDash \pi\}$, $\Pi_q \in 2^{\Pi}$, denote the set of all atomic propositions satisfied at q. A *trajectory* or *run* of T starting from q is an infinite sequence $r = r(1)r(2)r(3)\ldots$ with the property that $r(1) = q$, $r(i) \in Q$, and $(r(i), r(i+1)) \in \rightarrow$, for all $i \geq 1$. A trajectory $r = r(1)r(2)r(3)\ldots$ defines a *word* $w = w(1)w(2)w(3)\ldots$, where $w(i) = \Pi_{r(i)}$.

Next, we give a brief review of the propositional linear temporal logic LTL_{-X} [6]. A linear temporal logic LTL_{-X} formula over Π is recursively defined as follows: (i) Every atomic proposition π_i, $i = 1, \ldots, K$ is a formula, and (ii) If ϕ_1 and ϕ_2 are formulas, then $\phi_1 \vee \phi_2$, $\neg\phi_1$, $\phi_1 \mathcal{U} \phi_2$ are also formulas. The semantics of LTL_{-X} formulas are given over words of transition system T. Formally, the satisfaction of formula ϕ at position $i \in \mathbb{N}$ of word w, denoted by $w(i) \vDash \phi$, is defined recursively as follows: (1) $w(i) \vDash \pi$ if $\pi \in w(i)$, (2) $w(i) \vDash \neg\phi$ if $w(i) \nvDash \phi$, (3) $w(i) \vDash \phi_1 \vee \phi_2$ if $w(i) \vDash \phi_1$ or $w(i) \vDash \phi_2$, and (4) $w(i) \vDash \phi_1 \mathcal{U} \phi_2$ if there exist a $j \geq i$ such that $w(j) \vDash \phi_2$ and for all $i \leq k < j$ we have $w(k) \vDash \phi_1$. Finally, a word w satisfies an LTL_{-X} formula ϕ, written as $w \vDash \phi$, if $w(1) \vDash \phi$.

The symbols $\neg$ and $\vee$ stand for negation and disjunction. The Boolean constants $\top$ and $\bot$ are defined as $\top = \pi \vee \neg\pi$ and $\bot = \neg\top$. The other Boolean connectors $\wedge$ (conjunction), $\Rightarrow$ (implication), and $\Leftrightarrow$ (equivalence) are defined from $\neg$ and $\vee$ in the usual way. The *temporal operator* $\mathcal{U}$ is called the *until* operator. Formula $\phi_1 \mathcal{U} \phi_2$ intuitively means that (over a word) ϕ_2 will eventually become true and ϕ_1 is true until this happens. Two useful additional temporal operators, "eventually" and "always" can be defined as $\Diamond\phi = \top\mathcal{U}\phi$ and $\Box\phi = \phi\mathcal{U}\bot$, respectively. Formula $\Diamond\phi$ means that ϕ becomes eventually true, whereas $\Box\phi$ indicates that ϕ is true at all positions of w. More expressiveness can be achieved by combining the temporal operators. Examples include $\Box\Diamond\phi$ (ϕ is true infinitely often) and $\Diamond\Box\phi$ (ϕ becomes eventually true and stays true forever).

B. Rectangles and multi-affine functions

For $N \in \mathbb{N}$, an N-dimensional rectangle $R_N(a, b)$ in the Euclidean space $\mathbb{R}^N$ is characterized by two vectors $a = (a_1, \ldots, a_N)$ and $b = (b_1, \ldots, b_N)$ with the property that $a_i < b_i$ for all $i = 1, \ldots, N$:

$$R_N(a,b) = \prod_{i=1}^{N} [a_i, b_i] = \{x \in \mathbb{R}^N \mid \forall i \in \{1, \ldots, N\} : \; a_i \leq x_i \leq b_i\}. \quad (1)$$

Let $V_N(a,b) = \prod_{i=1}^{N} \{a_i, b_i\}$ be the set of vertices of $R_N(a,b)$, and $\mathcal{F}_N(a,b)$ the set of facets of $R_N(a,b)$. $\mathcal{F}_N(a,b)$ has $2N$ elements: for each $i \in \{1, \ldots, N\}$ the intersections of $R_N(a,b)$ with the hyperplanes $x_i = a_i$ or $x_i = b_i$ are facets of $R_N(a,b)$, with normal vectors $-e_i$ and $+e_i$ respectively, pointing out of the rectangle. Here $e_1, \ldots, e_N$ denote the standard basis of $\mathbb{R}^N$. For any facet $F \in \mathcal{F}_N(a,b)$, $\mathcal{V}(F)$ denotes the set of vertices of F.

Definition 1: A *multi-affine function* $h : \mathbb{R}^N \longrightarrow \mathbb{R}^q$ (with $N, q \in \mathbb{N}$), is a function that is affine in each of its variables, i.e. h is of the form

$$h(x_1, \ldots, x_N) = \sum_{i_1, \ldots, i_N \in \{0,1\}} c_{i_1,\ldots,i_N} x_1^{i_1} \cdots x_N^{i_N},$$

with $c_{i_1,\ldots,i_N} \in \mathbb{R}^q$ for all $i_1, \ldots, i_N \in \{0,1\}$, and using the convention that if $i_k = 0$, then $x_k^{i_k} \equiv 1$.

Multi-affine functions on multi-dimensional rectangles satisfy the following properties ([5]):

(i) *(Multi-affine functions are uniquely determined by their values at the vertices of a multi-dimensional rectangle)* For any function $g : V_N(a,b) \longrightarrow \mathbb{R}^q$ there exists a *unique* multi-affine function $h : \mathbb{R}^N \longrightarrow \mathbb{R}^q$ with the property that $h(v) = g(v)$ for all $v \in V_N(a,b)$. In particular, if $\xi_k : V_N(a,b) \longrightarrow \{0,1\}$ denotes the indicator function for the k-th component of a vertex v, i.e. $\xi_k(v) = 0$ if $v_k = a_k$, and $\xi_k(v) = 1$ if $v_k = b_k$, then h is given by

$$h(x_1, \ldots, x_N) = \sum_{v \in V_N(a,b)} \prod_{k=1}^{n} \left(\frac{x_k - a_k}{b_k - a_k}\right)^{\xi_k(v)} \left(\frac{b_k - x_k}{b_k - a_k}\right)^{1-\xi_k(v)} \cdot g(v). \quad (2)$$

(ii) In every point $x \in R_N(a,b)$ the value $h(x)$ of a multi-affine function is a convex combination of the values $\{h(v) \mid v \in V_N(a,b)\}$. Furthermore, if x belongs to a face of $R_N(a,b)$, then $h(x)$ is a convex combination of the values of h at the vertices of that face.

III. Problem Formulation

Definition 2: A *Rectangular Multi-Affine Hybrid System* (RMAHS) is a tuple

$$H = (Q, X_0, T, N, Inv, G, r, S, U) \quad (3)$$

where Q is a finite set of discrete states (or modes), $T \subseteq Q \times Q$ is a set of discrete transitions, $N : Q \rightarrow \mathbb{N}$ is a map giving the dimension N_q of the continuous state in each mode q, $Inv : Q \rightarrow 2^{\mathbb{R}^{N_q}}$ is the invariant map defined as $Inv(q) = R_{N_q}(a_q, b_q)$, $\bigcup_{q \in Q} Inv(q)$ is the set of continuous states, $X = \bigcup_{q \in Q} q \times Inv(q)$ is the total hybrid state space. G is a guard which associates a subset of $\mathcal{F}_{N_q}(a_q, b_q)$ to each transition $(q, q') \in T$. r is a reset map $r_{(q,q')} : R_{N_q}(a_q, b_q) \rightarrow R_{N_{q'}}(a_{q'}, b_{q'})$. Finally, S is an assignment of control systems to each invariant set in the form

$$S_q(x_q) = h_q(x_q) + B_q u, \quad (4)$$

where $h_q : \mathbb{R}^{N_q} \rightarrow \mathbb{R}^{N_q}$ is multi-affine, $B_q \in \mathbb{R}^{N_q \times m}$, and $u \in U$, where U is a polytope in $\mathbb{R}^m$ capturing the control bounds. The continuous evolution of the system in each N_q-dimensional rectangle $R_{N_q}(a_q, b_q)$ is given by

$$\dot{x}_q(t) = h_q(x_q(t)) + B_q u(t), \; x_q(t_0) = x_q^0 \quad (5)$$

In Definition 2, we assume that for every mode q, the guard sets consist of facets of the rectangle $Inv(q)$, in such a way that the guard sets cover the boundary of the rectangle and they do not overlap. Also, we assume that, through the reset map $r_{(q,q')}$, the initial state $x^0_{q'}$ in the new mode q' is uniquely determined by the final continuous state in the previous mode q. Since the explicit form of these reset maps does not play a role in the rest of the paper, they are not specified any further.

The semantics of the hybrid system in Definition 2 are given by its trajectories (q, x_q) as continuous time evolves. Explicitly, the system starts at time 0 from any initial state $(q_0, x^0_{q_0}) \in X_0$ and evolves in $R_{N_{q_0}}(a_{q_0}, b_{q_0})$ along the vector field (5) with $q = q_0$ until a guard is hit. If we denote this guard by $G(q_0, q_1)$, then the discrete transition (q_0, q_1) is taken, the continuous state is reset to $x^0_{q_1} \in R_{N_{q_1}}(a_{q_1}, b_{q_1})$ in accordance to $r_{(q_0,q_1)}$, and the procedure is reiterated. In this paper we are only interested in the discrete part of the trajectory:

Definition 3: A *word* generated by a trajectory $(q, x_q(t))$, $q \in Q$, $t \geq 0$ of H starting from $(q_0, x^0_{q_0})$ is an infinite sequence $w = w(1)w(2), \ldots$ satisfying $w(i) \in Q$ for all $i = 1, 2, \ldots$ and constructed inductively by the following three rules: (1) $w(1) = q_0$, (2) a symbol $w(i+1) \neq w(i)$, $i \geq 1$ is added to the sequence if $(w(i), w(i+1)) \in T$ and the continuous trajectory $x_{w(i)}$ initialized at $x^0_{w(i)}$ hits the guard $G(w(i), w(i+1))$ in finite time, without crossing other facets first, (3) an infinite number of symbols $w(i)$, $i \geq 1$ is added to the sequence if the continuous trajectory $x_{w(i)}$ initialized at $x^0_{w(i)}$ stays in $Inv(w(i))$ for all future times (without crossing any of the guards $G(w(i), q)$, with $(w(i), q) \in T$).

Informally, by Definition 3, the word generated by a trajectory of H is the enumeration of the modes reached by the trajectory, with infinitely many repetitions of a mode if that mode is reached and then never left.

Definition 4: The hybrid system H from Definition 2 satisfies an LTL_{-X} formula ϕ over Q if and only if all the words produced by all its trajectories satisfy the formula.

Problem 1: For an arbitrary LTL_{-X} formula ϕ over Q, find a set of initial states X_0 and a feedback control strategy $u \in U$ so that the closed loop hybrid system H satisfies formula ϕ.

The feedback control strategy will consist of an assignment of multi-affine state feedback controllers to each rectangle, with the possibility that more than one feedback controller is assigned to a given rectangle.

The solution to Problem 1 is presented in the next two sections. In Section IV, we focus on one mode, and derive sufficient conditions that guarantee that all trajectories of a multi-affine system leave the state rectangle through one or more a priori specified facets [1]. In the hybrid system H, this corresponds to the enabling of a guard corresponding to a transition. These conditions will lead in Section V to the construction of a *generator transition system*, with the property that all its words can be produced by H, independent of the value of the initial state in an invariant, and independent of the reset maps. A procedure very similar to LTL model checking will be used on the generator transition system to find runs satisfying the formula. These runs will eventually generate the feedback control strategy at the end of Section V.

[1]The results of this section are an extension of our previous work [5], in which we derived sufficient conditions for control to a facet. They may be considered as generalizations of the results in [11] to the multi-affine case.

IV. Control-to-facet problems

We focus on one mode q, and consider the multi-affine system

$$\dot{x}(t) = h(x(t)) + Bu(t), \qquad x(t_0) = x_0 \in R_N(a, b), \quad (6)$$

on the multi-dimensional rectangle $R_N(a, b)$, and with input $u \in U$. To limit the number of possible transitions from mode q to other modes, a feedback should be chosen in such a way, that the continuous state can only leave $R_N(a, b)$ through a facet that belongs to a specific subset $\mathcal{E}$ of the set $\mathcal{F}_N(a, b)$ of facets of $R_N(a, b)$, called the *admissible exit facets*.

Problem 2 (Control-to-facet): Let $\mathcal{E} \subset \mathcal{F}_N(a, b)$. Find an admissible multi-affine feedback $k : R_N(a, b) \longrightarrow U$ such that the multi-affine closed-loop system

$$\dot{x}(t) = h(x(t)) + Bk(x(t)), \qquad x(0) = x_0, \quad (7)$$

satisfies the following property: for every $x_0 \in R_N(a, b)$, solution $x(t, x_0)$ of (7) leaves rectangle $R_N(a, b)$ in finite time by crossing one of the facets $F \in \mathcal{E}$. If $\mathcal{E} = \varnothing$, all solutions $x(t, x_0)$ of (7) should remain in $R_N(a, b)$ for all $t \geq 0$.

Lemma 1: [5] Solutions $x(t, x_0)$ of closed-loop system (7) can not leave $R_N(a, b)$ by crossing one of the facets in $\mathcal{F}_N(a, b) \backslash \mathcal{E}$ if and only if

$$\forall F \in \mathcal{F}_N(a, b) \backslash \mathcal{E},\ \forall x \in F : \ n_F^T(h(x) + Bk(x)) \leq 0, \ (8)$$

with n_F the unit outward normal vector of facet F (i.e., apart from the sign, n_F is a unit vector).

The next result provides a sufficient condition to guarantee that all state trajectories leave the rectangle in finite time.

Lemma 2: Let $\dot{x} = h(x)$ be an autonomous multi-affine system on rectangle $R_N(a, b)$, and consider the following claims

(i) $\exists n \in \mathbb{R}^N \ \forall x \in R_N(a, b) : \ n^T h(x) > 0$,
(ii) For all $x_0 \in R_N(a, b)$, trajectory $x(t, x_0)$ leaves $R_N(a, b)$ in finite time,
(iii) $\forall x \in R_N(a, b) : \ h(x) \neq 0$.

Then (i) $\Longrightarrow$ (ii) $\Longrightarrow$ (iii).

Proof: (i) $\Longrightarrow$ (ii): Since $R_N(a, b)$ is a compact set, *(i)* guarantees that there exists a $c > 0$ such that the velocity in the direction of n is always larger than c. Hence $R_N(a, b)$ is left in finite time.

(ii) $\Longrightarrow$ (iii): Suppose $h(x_0) = 0$. Then $x(t, x_0) = x_0$ for all $t \geq 0$. This yields a contradiction. ■

For affine systems on polytopes, one may also prove that *(iii) $\Longrightarrow$ (i)* (see [12]). This implication is no longer valid in the multi-affine case, because the set $\{h(x) \mid x \in R_N(a, b)\}$ is not necessarily convex.

Next, like in [5], [11], [12], [17], the conditions in Lemmas 1 and 2 are restated in terms of linear inequalities at the vertices of the state rectangle.

Theorem 1: Consider a multi-affine system (6) on a multi-dimensional rectangle $R_N(a,b)$, with inputs from an input polytope $U \subset \mathbb{R}^m$. Let $\mathcal{E} \subset \mathcal{F}_N(a,b)$ be a set of admissible exit facets. For every vertex $v \in V_N(a,b)$, let $\mathcal{F}_v = \{F \in \mathcal{F}_N(a,b) \mid v \in \mathcal{V}(F)\}$ denote the set of all facets of $R_N(a,b)$ of which v is a vertex. For all $v \in V_N(a,b)$ we define the (possibly empty) input polytope

$$U_v := \{u \in U \mid n_F^T(h(v) + Bu) \leq 0 \text{ for all } F \in \mathcal{F}_v \backslash \mathcal{E}\}.$$

If $\mathcal{E} \neq \varnothing$, then Problem 2 is solvable if the following conditions are satisfied:

(i) $U_v \neq \varnothing$ for all $v \in V_N(a,b)$,

(ii) For all $v \in V_N(a,b)$ the input polytope U_v has a vertex w_v such that

$$0 \notin \mathrm{Conv}(\{h(v) + Bw_v \mid v \in V_N(a,b)\}),$$

where Conv denotes the convex hull. In particular, if for all $v \in V_N(a,b)$ inputs $u_v \in U_v$ are chosen in such a way that $0 \notin \mathrm{Conv}(\{h(v) + Bu_v \mid v \in V_N(a,b)\})$, then an admissible multi-affine state feedback solving Problem 2 is given by formula (2), with $g(v)$ replaced by u_v.

If $\mathcal{E} = \varnothing$, then Problem 2 is solvable if and only if $U_v \neq \varnothing$ for all $v \in V_N(a,b)$. In this case, every choice $g(v) \in U_v$ in formula (2) yields a multi-affine feedback solution.

Proof: (Case $\mathcal{E} \neq \varnothing$): Assume there exist $u_v \in U_v$, $(v \in V_N(a,b))$, satisfying condition *(ii)*. Let k denote the multi-affine function (2), with $g(v)$ replaced by u_v. Then k is an admissible multi-affine feedback, because $k(v) \in U$ for all $v \in V_N(a,b)$, and U is convex. Let $F \in \mathcal{F}_N(a,b) \backslash \mathcal{E}$. Then for all $v \in \mathcal{V}(F)$ we have $n_F^T(h(v) + Bk(v)) = n_F^T(h(v) + Bu_v) \leq 0$. Since the closed-loop dynamics $h(x) + Bk(x)$ is multi-affine, this implies that $n_F^T(h(x) + Bk(x)) \leq 0$ for all $x \in F$, and the closed-loop system cannot leave $R_N(a,b)$ by crossing facet F. Finally, since $0 \notin \mathrm{Conv}(\{h(v) + Bu_v \mid v \in V_N(a,b)\}) =: P$, there exists a hyperplane $n^T x = \alpha$ with $\alpha > 0$, that separates P and $\{0\}$. Since the closed-loop dynamics is multi-affine, this implies that $n^T(h(x) + Bk(x)) > \alpha > 0$ for all $x \in R_N(a,b)$. Hence, all state trajectories leave the state rectangle in finite time. Since facets $F \in \mathcal{F}_N(a,b) \backslash \mathcal{E}$ cannot be crossed, the state leaves $R_N(a,b)$ through an admissible exit facet $F \in \mathcal{E}$.

(Case $\mathcal{E} = \varnothing$): Sufficiency may be proved as above. For necessity we refer to [12], where a similar proof is given for the affine case. ■

Remark 1: The importance of Theorem 1 lies in the fact that the sufficient conditions for solvability of Problem 2 can be checked in a finite number of steps, using existing software on polyhedral sets (see e.g. [9], [14]). Furthermore, the theorem describes a constructive method for finding an admissible multi-affine feedback solution.

In [11], [12], the same problem has been solved for affine systems on simplices. In this situation the conditions of Theorem 1 turn out to be both necessary and sufficient.

V. Control Strategy

Definition 5: The generator transition system $T_g = (Q_g, Q_{g0}, \to_g, \Pi_g, \vDash_g)$ is defined by

- $Q_g = Q_{g0} = Q$,
- For all $q, q' \in Q$, $(q, q') \in \to_g$ if either $(q, q') \in T$ and there exists a solution to Problem 2 with $\mathcal{E} = G(q, q')$, or $q = q'$ and there exists a solution to Problem 2 with $\mathcal{E} = \emptyset$,
- $\Pi_g = Q$ and $(q, q') \in \vDash_g$ if and only if $q = q'$.

In other words, the states of the generator transition system T_g are the modes of H. Its set of transitions is the union of two sets of transitions. The first set is a subset of transitions of H for which multi-affine feedback controllers can be constructed so that all the continuous states in the invariant corresponding to the source discrete state can be driven to the corresponding guard in finite time. The second set is a set of self-transitions corresponding to the possibility of keeping all the continuous states in the corresponding invariant for all future times. Finally, the predicates associated with the states are the states themselves.

In the following, we use the notation $k_{qq'}(x)$ to denote the feedback controller corresponding to transition (q, q') of T_g.

To find runs of T_g satisfying an arbitrary LTL_{-X} formula ϕ over Q, we start by translating ϕ into a Büchi automaton $\mathcal{B}_\phi$. To this goal, we use the conversion algorithm described in [10] and its freely downloadable implementation LTL2BA. Then we take the (synchronous) product of T_g with $\mathcal{B}_\phi$ to obtain a product automaton $\mathcal{A}_{g,\phi}$ [13]. We use standard algorithms for graph traversing on $\mathcal{A}_{g,\phi}$ and eventually project back to find the desired runs of T_g. This approach is inspired by model checking algorithms, which are used to verify if a transition system satisfies a property expressed in terms of LTL. The difference is that a model checker constructs a Büchi automaton for the negation of the LTL formula and the product automaton is checked for emptiness (*i.e.* non-existence of accepted runs).

While we refer the reader to [15] for details, it is important to note that we consider only runs of T_g that have a special structure composed of one *prefix* and an infinite number of repetitions of a *suffix*. Note that this is not restrictive, since it can be proved [13] that, if there is an accepted run, then there is at least one accepted run with the above structure. Let $r_q = r_q(1)r_q(2)r_q(3)\ldots$, $r_q(j) \in Q$ denote the nonempty run of T_g starting from state q, *i.e.,* $r_q(1) = q$, $q \in Q_0$, where $Q_0 \subseteq Q$ is the set of indices of all nonempty runs. The set Q_0 is found by checking for existence of nonempty runs from each initial state of product automaton $\mathcal{A}_{g,\phi}$. The fact that r_q has the prefix-suffix structure can be formally written as: for any $q \in Q_0$, there exists n_p^q and n_s^q such that for any $j > n_p^q + n_s^q$, $r_q(j) = r_q((j - n_p^q - 1) \bmod n_s^q + n_p^q + 1)$. n_p^q and n_s^q are the number of states in prefix and suffix of r_q, respectively and thus the run r_q contains at most $n_p^q + n_s^q$ different states.

In [15] we also proved that, in a run r_q, $q \in Q_0$ of T_g, none of the states can be succeeded by itself, except for the state of a suffix of length one (case in which this state will be infinitely repeated). More formally, each run $r_q = r_q(1)r_q(2)r_q(3)\ldots$, $q \in Q_0$, satisfies the following property: $r_q(j) \neq r_q(j+1)$, $\forall j \in \mathbb{N} \setminus \{0\}$, $j \neq n_p^q + k\, n_s^q + 1$, $k \in \mathbb{N}$. Moreover, if $n_s^q \geq 2$, $r_q(j) \neq r_q(j+1)$, $\forall j \in \mathbb{N} \setminus \{0\}$.

We are now ready to provide a solution to Problem 1. The set of initial states is

$$X_0 = \bigcup_{q \in Q_0} q \times Inv(q) \tag{9}$$

where, as defined above, Q_0 is the set of all states of T_g from which there exists non-empty runs satisfying the formula. The control strategy is defined as follows:

Definition 6 (Control strategy): A control strategy for H corresponding to an LTL_{-X} formula ϕ is a tuple $C^\phi = (L, L_0, u, I, Rel)$, where:

- $L = \{l^q_{r_q(j)r_q(j+1)} \,|\, q \in Q, j \geq 1\}$ is its set of locations [2],
- $L_0 = \{l^q_{qr_q(2)}, q \in Q_0\}$ is the set of initial locations,
- $I(l^q_{r_q(j)r_q(j+1)}) = Inv(r_q(j))$ gives the invariant for each location,
- u is a map which assigns to each location $l^q_{r_q(j)r_q(j+1)}$ and continuous state $x_q \in I(l^q_{r_q(j)r_q(j+1)})$ a feedback controller $u(l^q_{r_q(j)r_q(j+1)}, x_q) = k_{r_q(j)r_q(j+1)}(x_q)$,
- $Rel \subseteq L \times L$, $Rel = \{(l^q_{r_q(j)r_q(j+1)}, l^q_{r_q(j+1)r_q(j+2)}), q \in Q_0, j \geq 1, r_q(j) \neq r_q(j+1)\}$

A location $l^q_{r_q(j)r_q(j+1)}$ corresponds to position j in run r_q, with r_q satisfying ϕ and determined as explained before. According to the structure of runs described above, the set of locations L is finite, even though the runs are infinite. A location $l^q_{r_q(j)r_q(j+1)}$ corresponds to driving all continuous states from rectangle $Inv(r_q(j))$ to the guard $G(r_q(j), r_q(j+1))$ in finite time if $(r_q(j), r_q(j+1)) \in T$, or to keeping the state of the system in $Inv(r_q(j))$ without hitting any guard for all times if $r_q(j) = r_q(j+1)$, by using the control $k_{r_q(j)r_q(j+1)}(x_q)$. Note that there can be several locations of C^ϕ mapped to the same invariant $Inv(q)$, $q \in Q$ of H. These can correspond to different runs of T_g passing through q or to states of the same run passing through q at different times and with different successors.

The semantics of the closed loop system (3), (5) with control strategy from Definition 6 is defined as follows: starting from $(q, x^0_q) \in X_0$ and location $l = l^q_{qr_q(2)} \in L_0$, feedback controller $u(l, x_q)$ is applied to system (3), (5) as long as the state $x_q \in I(l)$. When (and if) H takes a discrete transition corresponding to its semantics defined in Section III, then the location of C^ϕ is updated to l' according to $(l, l') \in Rel$ and the process continues.

We are now ready to provide a solution to Problem 1:

Theorem 2: The hybrid system H (equations (3), (5)), with feedback control strategy given by Definition 6 and set of initial states as in (9), satisfies the LTL_{-X} formula ϕ.

Proof: The proof follows from the construction of C^ϕ from Definition 6, the satisfaction of an LTL_{-X} formula by words generated by hybrid trajectories of H as in Definition 3, and from the sufficient conditions given by Theorem 1.

Let $(q, x_q(t))$ denote the trajectory of H, evolving under control law C^ϕ, and starting from an arbitrary initial state $(q_0, x^0_{q_0}) \in X_0$, with X_0 given by (9). From Definition 3, the corresponding word w has $w(1) = q_0$. The feedback control $k_{q_0 r_{q_0}(2)}(x_{q_0})$ is designed in accordance with Theorem 1.

If $r_{q_0}(2) = q_0$, then $k_{q_0 r_{q_0}(2)}(x_{q_0})$ is a solution of Problem 2 for rectangle $R_{N_{q_0}}(a_{q_0}, b_{q_0})$ with $\mathcal{E} = \varnothing$. From Definition 6, location $l^{q_0}_{q_0 q_0}$ cannot be left, so the control law does not change. Thus, $R_{N_{q_0}}(a_{q_0}, b_{q_0})$ is never left by the continuous trajectory, and Definition 3 implies that $w = q_0 q_0 \ldots$. As noted before, equality $r_{q_0}(2) = q_0$ holds only if $r_{q_0} = q_0 q_0 \ldots$, and because r_{q_0} satisfies ϕ, Definition 4 implies that H, under control strategy C^ϕ, also satisfies specification ϕ.

If $r_{q_0}(2) \neq q_0$, then $k_{q_0 r_{q_0}(2)}(x_{q_0})$ is a solution of Problem 2 for rectangle $R_{N_{q_0}}(a_{q_0}, b_{q_0})$ with $\mathcal{E} = G(q_0, r_{q_0}(2))$. Thus, the guard $G(q_0, r_{q_0}(2))$ is hit in finite time, the mode of H becomes $r_{q_0}(2)$ and C^ϕ updates its location to $l^{q_0}_{r_{q_0}(2) r_{q_0}(3)}$. From Definition 3, the symbol $w(2) = r_{q_0}(2)$ is added to word w and the process continues in the same manner. A similar reasoning can be used for any position in the run, and we conclude that the obtained word w is equal to r_{q_0}, which proves the theorem. ∎

Remark 2: It is possible that the trajectories of the closed loop system visit some states more than once, and have different velocities at the same continuous state at different times. Therefore, the obtained feedback controllers are in general time-variant.

Remark 3: Our approach to solving Problem 1 is obviously conservative. If the proposed control strategy does not yield a solution, this does not imply that there exists no solution. The conservativeness of our approach arises from different sources. First of all, we do not consider the influence of the reset maps. Secondly, in every state rectangle all initial states are treated in the same way, instead of allowing a partitioning of the state rectangle, in which each part may behave differently. Finally, Theorem 1 only provides sufficient conditions for existence of controllers, instead of equivalent conditions.

VI. Example

Consider a Rectangular Multi-Affine Hybrid System H with set of modes $Q = \{q_1, q_2, q_3, q_4\}$. The continuous state space corresponding to each mode is a two-dimensional rectangle, defined by $a_{q_1} = a_{q_3} = (0,0)$, $b_{q_1} = b_{q_3} = (2,1)$ and $a_{q_2} = a_{q_4} = (0,0)$, $b_{q_2} = b_{q_4} = (1,1)$. The facets of each rectangle q_i, $i = 1, \ldots, 4$ are denoted by $F^{q_i}_1 = \{(x_1, x_2) \in R_2(a_{q_i}, b_{q_i}) | x_2 = a_{q_i}(2)\}$, $F^{q_i}_2 = \{(x_1, x_2) \in R_2(a_{q_i}, b_{q_i}) | x_1 = b_{q_i}(1)\}$, $F^{q_i}_3 = \{(x_1, x_2) \in R_2(a_{q_i}, b_{q_i}) | x_2 = b_{q_i}(2)\}$ and $F^{q_i}_4 = \{(x_1, x_2) \in R_2(a_{q_i}, b_{q_i}) | x_1 = a_{q_i}(1)\}$. The guards of the hybrid system are given by $G(q_1, q_2) = \{F^{q_1}_2, F^{q_1}_3\}$, $G(q_1, q_4) = \{F^{q_1}_1, F^{q_1}_4\}$, $G(q_2, q_1) = \{F^{q_2}_1\}$, $G(q_2, q_3) = \{F^{q_2}_2, F^{q_2}_3\}$, $G(q_2, q_4) = \{F^{q_2}_4\}$, $G(q_3, q_1) = \{F^{q_3}_1, F^{q_3}_2\}$, $G(q_3, q_4) = \{F^{q_3}_3, F^{q_3}_4\}$, $G(q_4, q_1) = \{F^{q_4}_4\}$, $G(q_4, q_3) = \{F^{q_4}_1, F^{q_4}_2, F^{q_4}_3\}$.

The input u is in the set $U = [-1, 1]$, and the continuous dynamics of H are given by: $\dot{x}_{q_1} = (2 - 0.5x_1 + x_2 +$

[2]The locations of C^ϕ should not be confused with the discrete states (modes) of the hybrid system H. Note that other authors use 'locations', 'discrete states', and 'modes' interchangeably for the discrete states of a hybrid system.

$x_1x_2 + u$, $2 - 0.5x_1x_2 + 0.5u$), $\dot{x}_{q_2} = (-1 + 1.5x_1 + 0.5x_2 - 2x_1x_2 - 2u$, $1.5 - x_1 - 3x_2 + 1.5x_1x_2 + u)$, $\dot{x}_{q_3} = (1 - 1.5x_1 - 0.5x_2 + 0.75x_1x_2$, $2 - x_1 - 3x_2 + x_1x_2)$, $\dot{x}_{q_4} = (0.5 - 0.5x_1 - 1.5x_2 + x_1x_2 + u$, $0.5 + 1.5x_1 - x_2 - 2x_1x_2 - u)$. (For notational convenience we omitted the subscripts q_i from the continuous state components x_1 and x_2). An explicit definition of reset maps is not required in our approach; for simplicity we assume that each reset map positions the continuous state in the centroid of the rectangle corresponding to the new mode.

In order to construct the generator transition system T_g, we repeatedly apply Theorem 1 to each mode of H and for each guard set, including the empty one (for stay inside). For mode q_1 we obtain that transition to q_2 can be guaranteed for any initial continuous state in mode q_1 by the control law $k_{q_1q_2}(x) = 1$, $x \in R_2(a_{q_1}, b_{q_1})$, while transitions to q_4 and self loop in q_1 cannot be guaranteed, so they will not appear in T_g. Similarly, from mode q_2, transitions to q_1 and q_3 cannot be guaranteed, transition to q_4 is guaranteed by $k_{q_2q_4}(x) = 1$, while self loop in q_2 is possible under the control law $k_{q_2q_2}(x) = -1 + 2x_1 - x_1x_2$, $x \in R_2(a_{q_2}, b_{q_2})$. The control input doesn't affect the continuous dynamics inside mode q_3, but Theorem 1 reveals that $R_2(a_{q_3}, b_{q_3})$ is a sink for trajectories originating inside it, so T_g includes a self loop in q_3, with a dummy control law, *e.g.* $k_{q_3q_3}(x) = 0$, $x \in R_2(a_{q_3}, b_{q_3})$. Mode q_4 has transitions to q_1 (enabled by $k_{q_4q_1}(x) = -1 + 2x_2 - 2x_1x_2$), to q_3 (under control law $k_{q_4q_3}(x) = 1$), and self loop (with $k_{q_4q_4}(x) = -x_1 + x_2 - x_1x_2$), $x \in R_2(a_{q_4}, b_{q_4})$.

We impose the specification $\phi = q_1 \wedge \Diamond q_2 \wedge \Diamond q_4 \wedge \Diamond\Box q_2 \wedge \Box\neg q_3$, meaning that the hybrid system should start from mode q_1, then visit modes q_2 and q_4, and eventually converge to mode q_2, while always avoiding mode q_3. The specification can be met, and the corresponding discrete run of T_g has a four state prefix and a one state suffix in the form: $r_{q_1} = q_1q_2q_4q_1q_2q_2q_2\ldots$. Therefore, the controller C^ϕ has four locations ($l^{q_1}_{q_1q_2}$, $l^{q_1}_{q_2q_4}$, $l^{q_1}_{q_4q_1}$, $l^{q_1}_{q_2q_2}$), and the involved controllers are: $k_{q_1q_2}$ (while continuous state is in mode q_1), $k_{q_2q_4}$ (when mode q_2 is visited for the first time), $k_{q_4q_1}$ (in mode q_4) and $k_{q_2q_2}$ (when mode q_2 is visited for the second time). The control law in mode q_2 is time-variant, as noted in Remark 2.

VII. Concluding remarks

We studied the problem of controlling Rectangular Multi-Affine Hybrid Systems (RMAHS) from specifications given in terms of LTL_{-X} formulas over the set of their discrete states. We derived sufficient conditions for existence of controllers and a computationally efficient algorithm for automatic construction of such controllers. The conditions are stated in terms of sets of linear inequalities, and the involved computation consists of polyhedral set operations, construction of Büchi automata, and searches on graphs. In the future, we will apply these results to hybrid models of robotic and biological systems.

References

[1] R. Alur, T. A. Henzinger, G. Lafferriere, and G. J. Pappas. Discrete abstractions of hybrid systems. *Proceedings of the IEEE*, 88:971–984, 2000.

[2] G. Batt, D. Ropers, H. de Jong, J. Geiselmann, R. Mateescu, M. Page, and D. Schneider. Validation of qualitative models of genetic regulatory networks by model checking: analysis of the nutritional stress response in E.Coli. In *Thirteenth International Conference on Intelligent Systems for Molecular Biology*, pages 19–28, 2005.

[3] C. Belta. On controlling aircraft and underwater vehicles. In *IEEE International Conference on Robotics and Automation*, New Orleans, LA, pages 4905–4910, 2004.

[4] C. Belta, P. Finin, L.C.G.J.M. Habets, A. Halasz, M. Imielinksi, V.Kumar, and H. Rubin. Understanding the bacterial stringent response using reachability analysis of hybrid systems. In *Hybrid Systems: Computation and Control*, volume 2993 of *Lecture Notes in Computer Science*, pages 111–125. Springer-Verlag, Berlin, 2004.

[5] C. Belta and L.C.G.J.M. Habets. Control of a class of nonlinear systems on rectangles. *IEEE Transactions on Automatic Control*, 2006. to appear.

[6] E.M. Clarke, O. Grumberg, and D.A. Peled. *Model checking*. MIT Press, 1999.

[7] J.M. Davoren, V. Coulthard, N. Markey, and T. Moor. Non-deterministic temporal logics for general flow systems. In *Hybrid Systems: Computation and Control*, volume 2993 of *Lecture Notes in Computer Science*, pages 280–295. Springer-Verlag, Berlin, 2004.

[8] G. E. Fainekos, H. Kress-Gazit, and G. J. Pappas. Hybrid controllers for path planning: a temporal logic approach. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 4885–4890. Seville, Spain, 2005.

[9] K. Fukuda. Frequently asked questions in polyhedral computation. Report, ETH Zürich, 2000, www.ifor.math.ethz.ch/~fukuda/polyfaq/polyfaq.html.

[10] P. Gastin and D. Oddoux. Fast LTL to Büchi automata translation. In H. Comon G. Berry and A. Finkel, editors, *Proceedings of the 13th Conference on Computer Aided Verification (CAV'01)*, volume 2102 of *Lecture Notes in Computer Science*, pages 53–65. Springer-Verlag, Berlin, 2001.

[11] L.C.G.J.M. Habets, P.J. Collins, and J.H. van Schuppen. Reachability and control synthesis for piecewise-affine hybrid systems on simplices. *IEEE Trans. Aut. Control*, 51:938–948, 2006.

[12] L.C.G.J.M. Habets and J.H. van Schuppen. Control to facet problems for affine systems on simplices and polytopes with applications to control of hybrid systems. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 4175–4180. Seville, Spain, 2005.

[13] G. Holzmann. *The Spin Model Checker, Primer and Reference Manual*. Addison-Wesley, Reading, Massachusetts, 2004.

[14] B. Jeannet. Convex polyhedra library. Technical report, Verimag, Grenoble, 1999.

[15] M. Kloetzer and C. Belta. A fully automated framework for control of linear systems from LTL specifications. In *Hybrid Systems: Computation and Control*, volume 3927 of *Lecture Notes in Computer Science*, pages 333–347. Springer-Verlag, Berlin, 2006.

[16] M. Kloetzer and C. Belta. Hierarchical abstractions for robotic swarms. In *IEEE International Conference on Robotics and Automation*, Orlando, FL, pages 952–957, 2006.

[17] B. Roszak and M.E. Broucke. Necessary and sufficient conditions for reachability on a simplex. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 4706–4711. Seville, Spain, 2005.

[18] B. Shults and B.J. Kuipers. Proving properties of continuous systems: Qualitative simulation and temporal logic. *Artificial Intelligence*, 92(1-2):91–130, 1997.

[19] P. Tabuada and G. Pappas. Model checking LTL over controllable linear systems is decidable. In *Hybrid Systems: Computation and Control*, volume 2623 of *Lecture Notes in Computer Science*, pages 498–513. Springer-Verlag, Berlin, 2003.