Reachability Analysis of Multi-affine Systems

Marius Kloetzer and Calin Belta

Center for Information and Systems Engineering,
Boston University, 15 Saint Mary's Street,
Brookline, MA 02446
{kmarius, cbelta}@bu.edu

Abstract. We present a technique for reachability analysis of continuous multi-affine systems based on rectangular partitions. The method is iterative. At each step, finer partitions and larger discrete quotients are produced. We exploit some interesting convexity properties of multi-affine functions on rectangles to show that the construction of the discrete quotient at each step requires only the evaluation of the vector field at the set of all vertices of all rectangles in the partition and finding the roots of a finite set of scalar affine functions. The methodology promises to be easily extendable to rectangular hybrid automata with multi-affine vector fields and is expected to find important applications in analysis of biological networks and robot control.

1 Introduction

Reachability analysis is the problem of constructing the set of states reached by trajectories of a system originating in a given (possibly infinite dimensional) initial set. *Safety verification* is the problem of proving that a system does not have any trajectory from a given initial set to a given final (unsafe) set. For discrete systems with a finite number of states, these problems are *decidable*, *i.e.,* can be solved by a computer in a finite number of steps. For continuous and hybrid (*i.e.,* described by both continuous and discrete dynamics) systems, these problems are very difficult (in general undecidable) because of the uncountability of the state space.

One way to solve such problems for continuous and hybrid systems is to construct the set of states reached by the system, or an over-approximation of this set, by working directly in the continuous state space. Such methods are called *direct* and are not the subject of this paper. Our work can be included into the group of *indirect* methods, where a reachability problem for a continuous or hybrid system is mapped to a reachability problem for a finite state discrete system through *discrete abstractions*. The main idea in discrete abstractions is to iteratively partition the infinite dimensional continuous state space and produce partition quotients whose trajectories include the trajectories of the continuous or hybrid system. Such a discrete system is said to *simulate* the original system. If the converse is true, *i.e.,* the continuous or hybrid system simulate the discrete quotient, the two systems are called *bisimilar*, and the two

J. Hespanha and A. Tiwari (Eds.): HSCC 2006, LNCS 3927, pp. 348–362, 2006.

reachability problems become equivalent. Therefore, in this case, the reachability problem for a continuous or hybrid system becomes decidable.

The bisimulation relation was introduced in [1], formally defined for linear systems in [2], and for nonlinear systems in a categorical context in [3]. In [4], it has been shown that reachability is undecidable for a very simple class of hybrid systems. Several decidable classes have been identified though by restricting the continuous behavior of the hybrid system, as in the case of timed automata [5], multirate automata [6], [7], and rectangular automata [4], [8], or by restricting the discrete behavior, as in order-minimal hybrid systems [9, 10, 11]. All these decidable classes are too weak to represent continuous and hybrid system models encountered in practice. Then one might be satisfied with sufficient abstractions, when a discrete quotient that simulates the original system is enough to prove a safety property. But even finding the discrete quotient is not at all trivial. Related work focuses on partitioning using linear functions of the continuous variables, as in the method of predicate abstractions [12, 13], or using polynomial functions as in [13, 14]. However, to derive the transitions of the discrete quotient, one has to be able to either integrate the vector fields of the initial system [12], or use computationally expensive decision procedures such as quantifier elimination for real closed fields and theorem proving [13], which severely limit the dimensions of the problems that can be approached.

In this paper, we focus on formal analysis of continuous systems with multi-affine vector fields, *i.e,* affine in each variable, defined on rectangular regions of the Euclidean space. The main idea behind this work is, as in [15, 16, 17], to exploit the specific form of the vector field and the particular shape of the invariant to infer reachability properties of infinite uncountable sets of states from properties verified by a finite set of states. Specifically, in [18], we proved that a multi-affine function is uniquely determined by its values at the vertices of a rectangle and its restriction to the rectangle is a convex combination of these values. In this paper, we use this result to develop a reachability analysis algorithm for multi-affine systems by iteratively constructing finer and finer discrete quotients.

Even though the abstraction procedure in this paper falls into the more general framework of [13], we show that if more structure is allowed, then reachabiltity and safety verification questions can be answered with much less computation. The calculation of the discrete quotient at a given iteration involves only finding the roots of scalar affine functions and evaluation of multi-affine functions at a finite number of points. This will allow us to approach much larger problems, as usually found in analysis of bio-molecular networks, where the multi-affine structure appears naturally when chemical reactions with unitary stoichiometric coefficients are modelled using mass action kinetics. Multi-affine dynamics are also found in other systems, including the celebrated Euler's equations for angular velocity of rotation of rigid bodies, the equations of motion of translating and rotating rigid bodies with rotation parameterized by quaternions [19], Volterra [20], and Lotka-Volterra equations [21].

2 Continuous Systems and Discrete Quotients

Definition 1 (Continuous system). *We represent a continuous dynamical system as a pair*

$$CS = (X, f), \tag{1}$$

where $X \subseteq \mathbb{R}^n$, $n \in \mathbb{N}$ is its continuous state space and f is a smooth vector field on X, i.e., the state $x \in X$ of system (1) evolves according to $\dot{x} = f(x)$.

We assume that X is a connected subset of $\mathbb{R}^n$ and introduce a *set partition* of X by defining the *abstraction map*: $abs : X \to L$, where L is a finite set of labels for all the elements in the partition. Let *con* be the *concretization map* of the partition induced by *abs*: $con : L \to X$, $con(l) = \{x \in X | abs(x) = l\}$.

In other words, for $l \in L$, we use $con(l) \subseteq X$ to denote the set of all $x \in X$ in the partition element with label l. Since *abs* induces a partition and *con* is its concretization map, we have $\bigcup_{l \in L} con(l) = X$ and $con(l) \bigcap con(l') = \emptyset$, for all $l, l' \in L$, $l \neq l'$. We use $con(l) \sim con(l')$, or simply $l \sim l'$ to denote adjacency of regions $con(l)$ and $con(l')$. For simplicity of notation, we use $con(I)$ to denote $\bigcup_{l \in I} con(l)$, $I \subset L$. For an arbitrary $I \subset L$, we denote by $\text{Post}(con(I))$ the set of all states in X reached by the trajectories of (1) originating in $con(I)$, for all times $t \geq 0$. The reachability problem for CS can be formulated as follows:

Problem 1 (Reachability). For an arbitrary $I \subset L$, determine $\text{Post}(con(I))$.

The safety verification problem for CS is the problem of deciding if (1) has trajectories between arbitrary regions in the partition induced by the map *abs*:

Problem 2 (Safety). Given $I, F \subset L$ with $I \bigcap F = \emptyset$, determine the truth value of the following assertion:

$$\text{Post}(con(I)) \bigcap con(F) = \emptyset \tag{2}$$

In a particular application, $con(I)$ corresponds to a set of states around initial or operating points of a system CS, while $con(F)$ might represent unsafe regions.

Note that our definition of 'Post' operator implies that the reachability and safety problems we are dealing with are time-abstract. It is obvious to see that the solution to Problem 1 immediately gives a solution to Problem 2, provided that we can calculate the intersection in Eqn. (2). However, in order to solve Problem 2, it is not necessary to solve Problem 1 - it is enough to construct an over-approximation of $\text{Post}(con(I))$ that has empty intersection with $con(F)$. To construct over-approximations of $\text{Post}(con(I))$, we use *discrete quotients*:

Definition 2 (Discrete quotient). *A discrete quotient of CS induced by the partition map 'abs' is a finite state transition system DS described by the pair*

$$DS = (L, T), \tag{3}$$

where L is the set of labels produced by the abstraction map 'abs', and $T \subseteq L \times L$ is a set of transitions satisfying the following property:

$$(l, l') \in T \text{ if } l \sim l' \text{ and there exist } t_1, t_2 \geq 0,\ t_1 < t_2 \text{ and a trajectory } x(t) \text{ of } CS \text{ such that } x(t_1) \in con(l),\ x(t_2) \in con(l') \text{ and } x(t) \in (con(l)) \cap con(l'),\ \forall\, t \in [t_1, t_2]. \tag{4}$$

As before, for $I \subset L$, we denote by $\mathrm{Post}(I) \subseteq L$ the set of all discrete states that can be reached from I by DS. More formally, $\mathrm{Post}(I) = \bigcup_{l \in I} \mathrm{Post}(l)$.

Note that we use the same operator 'Post' for both CS and DS, with the observation that they are easily distinguished by their arguments.

From (4) it follows that

$$\mathrm{Post}(con(I)) \subseteq con(\mathrm{Post}(I)) \tag{5}$$

Eqn. (5) implies that, if the transitions (4) of a discrete quotient (3) can be computed, then an over-approximation $con(\mathrm{Post}(I))$ of $\mathrm{Post}(con(I))$ can be easily determined by a search on the transition system (3), which is a decidable problem. If $\mathrm{Post}(I) \bigcap F = \emptyset$ (equivalent with $con(\mathrm{Post}(I)) \cap con(F) = \emptyset$, since $con(L)$ is a partition of X), then the truth value of (2) is TRUE. Otherwise, we cannot answer Problem 2, and a less conservative discrete quotient is necessary.

There are two sources of conservativeness in the definition of DS. The first comes from the fact that, according to (4), there might exist a transition $(l, l') \in T$ even if CS does not have a trajectory from $con(l)$ to $con(l')$. A more correct definition of the discrete quotient should have 'if and only if' instead of 'if' in Eqn. (4). This would make CS and DS equivalent from the point of view of reachability of adjacent regions in one step. However, even in this case, there is a second source of conservativeness, which comes from lack of transitivity in the following sense: if $(l, l') \in T$ and $(l', l'') \in T$, which implies that l, l', l'' is a trajectory of DS, this does not imply that CS has a trajectory from $con(l)$ to $con(l')$ and to $con(l'')$, simply because it is possible that all trajectories that go from $con(l)$ to $con(l')$ escape to a region $con(l''')$, with $l''' \neq l''$. The conservativeness is completely eliminated, *i.e.,* CS and DS are equivalent with respect to reachability properties, if and only if, in (4), the 'if' statement is replaced by 'if and only if', and all initial states in $con(l)$ flow in finite time to $con(l')$ under the dynamics of CS.

As outlined in Section 1, finding such non-conservative discrete quotients of continuous systems is an extremely hard problem. Moreover, even finding discrete quotients with 'if and only if' in Eqn. (4) is very difficult. In this paper, we use the relaxed Definition 2 of a discrete quotient to construct less and less conservative over-approximations $con(\mathrm{Post}(I))$ for the solutions to Problems 1 and 2. Formally, we define a refinement of a discrete quotient as follows:

Definition 3 (Refinement). *For a given continuous system CS, a discrete quotient $\overline{DS} = (\bar{L}, \bar{T})$ induced by $\overline{abs} : X \to \bar{L}$ refines a discrete quotient $DS = (L, T)$ induced by $abs : X \to L$ if $|\bar{L}| > |L|$ and the following conditions hold:*

(i) For any $l \in L$, there exists $\bar{I} \subset \bar{L}$ with $|\bar{I}| \geq 1$ so that $\overline{con}(\bar{I})$ is a partition of $con(l)$. Any $\bar{l} \in \bar{I}$ is said to refine $l \in L$, and we denote this by $\bar{l} \leq l$.

(ii) *For any* $\bar{l}, \bar{l}' \in \bar{L}$ *with* $(\bar{l}, \bar{l}') \in \bar{T}$, *if there exist* $l, l' \in L$, $l \neq l'$, *so that* $\bar{l} \leq l$ *and* $\bar{l}' \leq l'$, *then* $(l, l') \in T$.

(iii) *There exist* $l, l' \in L$ *with* $(l, l') \in T$ *and* $\bar{l}, \bar{l}' \in \bar{L}$ *with* $\bar{l} \sim \bar{l}'$, $\bar{l} \leq l$, $\bar{l}' \leq l'$, *and* $(\bar{l}, \bar{l}') \notin \bar{T}$.

In other words, (i) states that each region in the partition produced by *abs* is further partitioned by $\overline{abs}$. Note that, since $|\bar{L}| > |L|$, at least one region $con(l)$ is strictly partitioned. Condition (ii) requires that the finer quotient $\overline{DS}$ can only have transitions between states refining states connected by transitions in the coarser quotient DS and between states refining the same state of DS. Conditions (i) and (ii) will guarantee that the over-approximation $con(\mathrm{Post}(I))$ as in Eqn. (5) does not grow through refinement. Finally, (iii) means that there exist at least one pair of states connected in the coarser DS for which refinement determines two disconnected states in the finer description $\overline{DS}$.

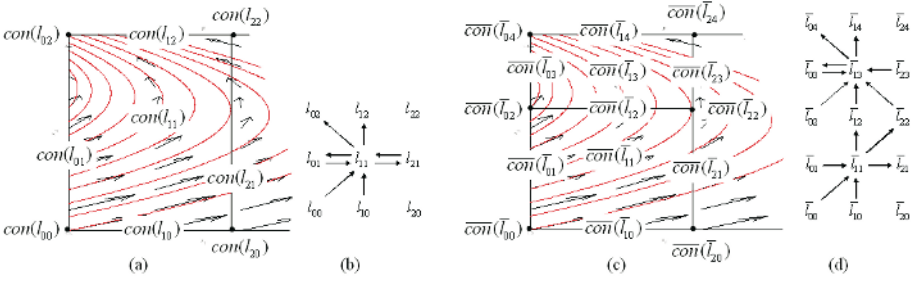

Fig. 1. Discrete quotients for a vector field $f = (f_1, f_2)$, $f_1 = 2 - x_1 x_2$, $f_2 = 1 + x_2 - x_1 x_2$ in a rectangular region $[1.5, 1.56] \times [1.1, 1.42]$ in plane. An initial partition and the corresponding discrete quotient are shown in (a) and (b), respectively. A finer partition is shown in (c), and the corresponding discrete quotient (d) refines the initial one (b). The regions of partitions are "open" rectangles of dimension 0 (points), 1 (open line segments), and 2 (rectangles without boundaries). The transitions of the discrete quotients correspond to 'if and only if' in Eqn. (4).

An example is given in Fig. 1, where an initial partition $\bigcup_{i,j=0,1,2} con(l_{ij})$ of a 2-dimensional rectangle (containing its boundaries) is refined to $\bigcup_{i=0,1,2;j=0,\ldots,4} \overline{con}(\bar{l}_{ij})$. It is easy to see that condition (ii) of Definition 3 is satisfied, *i.e.,* no "new" transitions are added. As it can be seen in Fig. 1(c), the refinement is achieved by "cutting" with a horizontal line where the f_1 component of the vector field becomes zero on the vertical open segment $con(l_{21})$. This leads to a partition $\overline{con}(\bar{l}_{13}), \overline{con}(\bar{l}_{12}), \overline{con}(\bar{l}_{11})$ of $con(l_{11})$ and a partition $\overline{con}(\bar{l}_{23}), \overline{con}(\bar{l}_{22}), \overline{con}(\bar{l}_{21})$ of $con(l_{21})$. In the finer quotient, it can be seen for example that there is no transition from $\bar{l}_{21}$ to $\bar{l}_{11}$ and from $\bar{l}_{13}$ to $\bar{l}_{23}$, even though the coarser quotient had transitions between l_{11} and l_{21} in both directions (condition (iii)).

Condition (iii) in Definition 3 is a necessary condition for strict shrinking of the over-approximation $con(\mathrm{Post}(I))$. However, it is not sufficient. Indeed, for adjacent regions $\bar{l} \sim \bar{l}'$, if CS does not have trajectories penetrating directly from

$\overline{con}(\bar{l})$ to $\overline{con}(\bar{l}')$, this does not mean that $Post(\overline{con}(\bar{l})) \bigcap \overline{con}(\bar{l}') = \emptyset$. Trajectories originating in $\overline{con}(\bar{l})$ can loop around and eventually hit $\overline{con}(\bar{l}')$.

These ideas are formalized in Proposition 1. Due to the space constraints, the proof of Proposition 1 is not included here, and it can be found in [22].

Proposition 1 (Conservativeness reduction by refinement). *If* $\overline{DS} = (\bar{L}, \bar{T})$ *refines* $DS = (L, T)$*, and* $I \subset L$*,* $\bar{I} \subset \bar{L}$ *with the property that* $\overline{con}(\bar{I})$ *is a partition of* $con(I)$*, then we have*

$$\mathrm{Post}(con(I)) = \mathrm{Post}(\overline{con}(\bar{I})) \subseteq \overline{con}(\mathrm{Post}(\bar{I})) \subseteq con(\mathrm{Post}(I)) \tag{6}$$

Moreover, if (iii) from Definition 3 is replaced by:

(iii)' There exist $l, l' \in L$ *with* $(l, l') \in T$ *and* $\bar{l}' \in \bar{L}$ *with* $\bar{l}' \leq l'$*, and* $\bar{l}' \notin \mathrm{Post}(\bar{l})$*,* $\forall \bar{l} \in \bar{L}$*,* $\bar{l} \leq l$

and $l \in (I \cup \mathrm{Post}(I))$*, then the last inclusion relation in (6) is strict,* i.e.*, the over-approximation* $con(\mathrm{Post}(I))$ *as in Eqn. (5) strictly shrinks through refinement.*

Remark 1 (Simulation and bisimulation). Both CS and DS defined above can be embedded into transition systems [2, 23] with set of observables L. In this framework, DS given by Definition 2 is said to simulate CS. When both sources of conservativeness mentioned above are eliminated, then CS simulates DS as well, and they are called bisimilar. The interested reader can refer to [1, 2, 23] for formal definitions of simulation and bisimulation relations.

In this paper, we assume that X is a full dimensional "closed" rectangle in $\mathbb{R}^n$ and the vector field f is multi-affine, *i.e.,* affine in each state component. We use iterative partitions of X into "open" rectangles and some interesting convexity properties of multi-affine functions on rectangles to calculate discrete quotients according to Definitions 2 and 3 and provide a solution to Problem 2 and a conservative solution to Problem 1. As it will be seen, we cannot guarantee the sufficient condition (iii)' for strict shrinking at each step of the refinement. Instead, we satisfy the necessary condition (iii), with the "hope" that the conservativeness is strictly reduced.

3 Rectangles and Multi-affine Functions

Two vectors $a = (a_1, \ldots, a_n) \in \mathbb{R}^n$ and $b = (b_1, \ldots, b_n) \in \mathbb{R}^n$ with the property that $a_i < b_i$ for all $i = 1, \ldots, n$ determine a set of 3^n rectangles in $\mathbb{R}^n$:

$$\mathcal{R}(a, b) = \{R_{(l_1,\ldots,l_n)},\ l_i \in \{0, 1, 2\},\ i = 1, \ldots, n\} \tag{7}$$

where each rectangle $R_{(l_1,\ldots,l_n)}$, $l_i \in \{0, 1, 2\}$, $i = 1, \ldots, n$ is defined by

$$\begin{aligned} R_{(l_1,\ldots,l_n)} = \{x = (x_1, \ldots, x_n) \in \mathbb{R}^n \mid x_i = a_i \text{ if } l_i = 0,\\ a_i < x_i < b_i \text{ if } l_i = 1,\ x_i = b_i \text{ if } l_i = 2,\ i = 1, \ldots, n\} \end{aligned} \tag{8}$$

We define the order m of a rectangle $R_{(l_1,\ldots,l_n)}$ as being the number of '1' entries in its label $(l_1,\ldots,l_n)$. The number of m - order rectangles in $\mathcal{R}(a,b)$ is $2^{n-m}n!/((n-m)!m!)$. As particular cases, there is only one n - order (full dimensional) rectangle $R_{(1,\ldots,1)}$, and 2^n 0 - order rectangles, or *vertices* $R_{(l_1,\ldots,l_n)}$, $l_i \in \{0,2\}$, $i=1,\ldots,n$. For a given rectangle $R_{(l_1,\ldots,l_n)}$, we can define

$$\begin{array}{c}\mathcal{L}R_{(l_1,\ldots,l_n)} = \{R_{(l'_1,\ldots,l'_n)} \in \mathcal{R}(a,b) \mid \\ (l'_1,\ldots,l'_n) \neq (l_1,\ldots,l_n) \wedge l'_i = l_i \text{ if } l_i \in \{0,2\}\}\end{array} \tag{9}$$

The set of vertices corresponding to $R_{(l_1,\ldots,l_n)}$ is a subset of $\mathcal{L}R_{(l_1,\ldots,l_n)}$ defined by

$$\begin{array}{c}\mathcal{V}R_{(l_1,\ldots,l_n)} = \{R_{(l'_1,\ldots,l'_n)} \in \mathcal{R}(a,b) \mid (l'_1,\ldots,l'_n) \neq (l_1,\ldots,l_n) \wedge \\ l'_i = l_i \text{ if } l_i \in \{0,2\} \wedge l'_i \in \{0,2\} \text{ if } l_i = 1\}\end{array} \tag{10}$$

If the order of $R_{(l_1,\ldots,l_n)}$ is m, there are $3^m - 1$ rectangles in $\mathcal{L}R_{(l_1,\ldots,l_n)}$, all of order less than or equal to $m-1$, and 2^m vertices (0-order rectangles) in $\mathcal{V}R_{(l_1,\ldots,l_n)}$. We call the rectangles defined by (8) *open* rectangles, with the observation that, except for $R_{(1,\ldots,1)}$, they are not open sets in $\mathbb{R}^n$. If all '$<$' in (8), if any, are replaced by '$\leq$', then $R_{(l_1,\ldots,l_n)}$ becomes *closed*, and is denoted by $\bar{R}_{(l_1,\ldots,l_n)}$. It is easy to see that $\bar{R}_{(l_1,\ldots,l_n)} = R_{(l_1,\ldots,l_n)} \cup \mathcal{L}R_{(l_1,\ldots,l_n)}$. For a closed rectangle $\bar{R}$, sets $\mathcal{L}\bar{R}$ and $\mathcal{V}\bar{R}$ are defined as in (9,10) by replacing R with $\bar{R}$. It follows that the sets of vertices of open and closed rectangles are identical, *i.e.,* $\mathcal{V}R = \mathcal{V}\bar{R}$. Therefore we will use $\mathcal{V}R$ for the set of vertices of $\mathcal{V}\bar{R}$.

Definition 4 (Multi-affine function). *A multi-affine function $f : \mathbb{R}^n \longrightarrow \mathbb{R}^p$ (with $p \in \mathbb{N}$) is a polynomial in the indeterminates $x_1,\ldots,x_n$ with the property that the degree of f in any of the variables is less than or equal to 1. Stated differently, f has the form:*

$$f(x_1,\ldots,x_n) = \sum_{i_1,\ldots,i_n \in \{0,1\}} c_{i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}, \tag{11}$$

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

The following proposition is proved in [18]:

Proposition 2. *A multi-affine function is uniquely determined by its values at the vertices $\mathcal{V}R_{(1,\ldots,1)}$ of a full dimensional closed rectangle $\bar{R}_{(1,\ldots,1)}$. Its restriction to the rectangle is a convex combination of the values at the vertices and has the following form:*

$$\begin{array}{c}f|_{\bar{R}_{(1,\ldots,1)}}(x_1,\ldots,x_n) = \sum_{(v_1,\ldots,v_n)\in\mathcal{V}R_{(1,\ldots,1)}} \\ \prod_{k=1}^n \left(\frac{x_k-a_k}{b_k-a_k}\right)^{\xi(v_k)} \left(\frac{b_k-x_k}{b_k-a_k}\right)^{1-\xi(v_k)} f(v_1,\ldots,v_n),\end{array} \tag{12}$$

where $\xi_k : \{a_1,\ldots,a_n,b_1,\ldots,b_n\} \to \{0,1\}$ is an indicator function defined by: $\xi_k(a_k) = 0$, $\xi_k(b_k) = 1$, $k = 1,\ldots,n$.

Since a multi-affine function remains multi-affine if some of its arguments are kept constant, Proposition 2 is true when a multi-affine function is restricted to a lower order closed rectangle, when Eqn. (12) becomes:

$$f|_{\bar{R}_{(l_1,\ldots,l_n)}}(x_1,\ldots,x_n) = \sum_{(v_1,\ldots,v_n)\in\mathcal{V}R_{(l_1,\ldots,l_n)}} \prod_{k,l_k=1} \left(\frac{x_k-a_k}{b_k-a_k}\right)^{\xi(v_k)} \left(\frac{b_k-x_k}{b_k-a_k}\right)^{1-\xi(v_k)} f(v_1,\ldots,v_n), \tag{13}$$

Note that $f|_{\bar{R}_{(l_1,\ldots,l_n)}}(x_1,\ldots,x_n)$ is obtained from $f|_{\bar{R}_{(1,\ldots,1)}}(x_1,\ldots,x_n)$ by setting $x_i = a_i$ for $l_i = 0$ and $x_i = b_i$ for $l_i = 2$, $i = 1,\ldots,n$.

A straightforward corollary of Proposition 2 can be stated as (the proof can be found in [22]):

Corollary 1. *If f is a scalar multi-affine function ($p = 1$ in Definition 4) and $R_{(l_1,\ldots,l_n)}$ is an open rectangle of arbitrary order, then we have:*

(a) $f(x) > 0$ everywhere in $R_{(l_1,\ldots,l_n)}$ if and only if $f(v) \geq 0$ for all $v \in \mathcal{V}R_{(l_1,\ldots,l_n)}$, and there exists at least one $v \in \mathcal{V}R_{(l_1,\ldots,l_n)}$ for which $f(v) > 0$.
(b) $f(x) < 0$ everywhere in $R_{(l_1,\ldots,l_n)}$ if and only if $f(v) \leq 0$ for all $v \in \mathcal{V}R_{(l_1,\ldots,l_n)}$, and there exists at least one $v \in \mathcal{V}R_{(l_1,\ldots,l_n)}$ for which $f(v) < 0$.
(c) $f(x) = 0$ everywhere in $R_{(l_1,\ldots,l_n)}$ if and only if $f(v) = 0$ for all $v \in \mathcal{V}R_{(l_1,\ldots,l_n)}$.
(d) There exist $x, x' \in R_{(l_1,\ldots,l_n)}$ with $f(x) > 0$ and $f(x') < 0$ if and only if there exist $v, v' \in \mathcal{V}R_{(l_1,\ldots,l_n)}$ with $f(v) > 0$ and $f(v') < 0$.

4 Reachability Analysis of Multi-affine Systems

We now have all the necessary background to consider Problems 2 and 1 for a continuous system CS (Definition 1) , whose continuous state space is a closed rectangle in $\mathbb{R}^n$ defined by $a = (a_1,\ldots,a_n) \in \mathbb{R}^n$ and $b = (b_1,\ldots,b_n) \in \mathbb{R}^n$, $a_i < b_i$ for all $i = 1,\ldots,n$:

$$X = \{x = (x_1,\ldots,x_n) \in \mathbb{R}^n \,|\, a_i \leq x_i \leq b_i,\ i = 1,\ldots,n\}, \tag{14}$$

and whose vector field f is multi-affine as in Definition 4 (with $p = n$).

We first define a partition of X into open rectangles, which gives the states of the discrete quotient DS (Definition 2). We then define the transitions of DS and a refinement procedure according to Definition 3. Finally, we collect all the results in an iterative algorithm for safety verification of multi-affine systems. Due to the space constraints, we give just some informal explanations of the involved algorithms, and we refer to [22] for pseudocodes.

4.1 The States of the Discrete Quotient

We assume that each axis Ox_i, $i = 1,\ldots,n$ is divided into $n_i \geq 1$ intervals by the points $\theta_0^i < \theta_1^i < \ldots < \theta_{n_i}^i$. This induces a partition of X into $\prod_{i=1}^n (2n_i + 1)$

open rectangles. Using the same idea as in Section 3, we label the rectangles with n - uples $(l_1, \ldots, l_n)$ by defining an abstraction map as follows:

$$abs(x_1, \ldots, x_n) = (l_1, \ldots, l_n) \tag{15}$$

where, for each $i = 1, \ldots, n$ and $j_i = 0, 1, \ldots, n_i$,

$$l_i = 2j_i, \text{ if } x_i = \theta^i_{j_i} \quad , \quad l_i = 2j_i - 1, \text{ if } \theta^i_{j_i - 1} < x_i < \theta^i_{j_i} \tag{16}$$

Remark 2. The connection with the work in [13] can be seen as follows: the polynomials $x_i - \theta^i_{j_i}$, $j_i = 0, \ldots, n_i$, $i = 1, \ldots, n$ define a set of discrete variables, which generate the set L when interpreted over the set of symbols $\{pos, neg, zero\}$ (with the obvious significance). In this representation, each discrete state $l \in L$ is a word of length $\sum_{i=1}^{n} n_i + n$ over the set $\{pos, neg, zero\}$, and the cardinality of L becomes $|L| = 3^{\sum_{i=1}^{n} n_i + n}$. However, in our definition (15), $|L| = \prod_{i=1}^{n}(2n_i + 1)$. The dramatic reduction in the number of discrete states comes form the fact that, in the rectangular partition, infeasible combinations of polynomial interpretations are automatically eliminated.

As defined in Section 3, the number m of odd entries in $l = (l_1, \ldots, l_n)$ is the order of the rectangle. Moreover, $con(l)$ is an open m - rectangle in X. From now on, when we refer to rectangles we mean open rectangles. If all l_i's are odd, then $con(l)$ is a (full dimensional) n - order rectangle and if all l_i's are even, then $con(l)$ is a point (vertex), or 0 - order rectangle. Inspired by this observation, we define the *order* of a discrete state l as the number of its odd entries.

4.2 The Transitions of the Discrete Quotient

Before we start constructing the set T of transitions from all discrete states $l \in L$, note that, because of the rectangular partition, it is easy to identify a subset of L where transitions are possible, so we don't have to explore the whole L in search for successors. Let

$$\begin{gathered}\mathcal{H}(l) = \{l' = (l'_1, \ldots, l'_n) \in L \,|\, l' \neq l \wedge \\ l'_i = l_i \, if \, l_i \; odd \wedge l'_i \in \{l_i - 1, l_i, l_i + 1\} \, if \, l_i \; even\}\end{gathered} \tag{17}$$

$$\begin{gathered}\mathcal{L}(l) = \{l' = (l'_1, \ldots, l'_n) \in L \,|\, l' \neq l \wedge \\ l'_i = l_i \, if \, l_i \; even \wedge l'_i \in \{l_i - 1, l_i, l_i + 1\} \, if \, l_i \; odd\}\end{gathered} \tag{18}$$

Note that, if l is an m - order discrete state, then all the discrete states in $\mathcal{H}(l)$ are of order strictly greater than m and all the discrete states in $\mathcal{L}(l)$ are of order strictly less than m. For a m - order discrete state $l = (l_1, \ldots, l_n)$, $1 \leq l_i \leq 2n_i - 1$, the cardinality of $\mathcal{H}(l)$ and $\mathcal{L}(l)$ are $3^{n-m} - 1$ and $3^m - 1$, respectively. Given $l \in L$, it is only possible to have discrete transitions towards discrete states in $\mathcal{H}(l) \cup \mathcal{L}(l)$. For a state l with order $m \geq 1$, let $\mathcal{V}(l)$ denote the set of labels of vertices of $con(l)$. Formally,

$$\begin{gathered}\mathcal{V}(l) = \{l' = (l'_1, \ldots, l'_n) \in L \,|\, l' \neq l \wedge \\ l'_i = l_i \, if \, l_i \; even \wedge l'_i \in \{l_i - 1, l_i + 1\} \, if \, l_i \; odd\}\end{gathered} \tag{19}$$

Before adding discrete transitions to complete the discrete system DS, we assign a *signature* to each discrete state $l \in L$.

Definition 5 (Signature of a discrete state). *For a discrete location* $l = (l_1, \ldots, l_n) \in L$*, the signature* $s(l) = (s_1(l), \ldots, s_n(l))$ *is a n-uple over the four-valued domain* $\{po, ne, ze, in\}$ *(*i.e., *positive, negative, zero, indefinite) with the following significance, for all* $i = 1, \ldots, n$*:*

- $s_i(l) = po$*, if* $f_i(x) > 0$*,* $\forall x \in con(l)$
- $s_i(l) = ne$*, if* $f_i(x) < 0$*,* $\forall x \in con(l)$
- $s_i(l) = ze$*, if* $f_i(x) = 0$*,* $\forall x \in con(l)$
- $s_i(l) = in$*, if* $\exists x \in con(l)$ *so that* $f_i(x) > 0$ *and* $\exists x \in con(l)$ *so that* $f_i(x) < 0$

where $f = (f_1, \ldots, f_n)$ *is the vector field of* CS*.*

The first and second cases correspond to the situation when $con(l)$ has an empty intersection with $f_i(x) = 0$. In the third case, $con(l)$ coincides with $f_i(x) = 0$ or $f_i(x) = 0$ contains $con(l)$. In the fourth, there is an intersection between $con(l)$ and $f_i(x) = 0$. The four cases from Definition 5 cover all possible choices for vector field f of CS.

Determining the signatures for 0 - order discrete states, *i.e.,* $l = (l_1, \ldots, l_n) \in L$ with all l_i even, is easy, because $con(l)$ is a point in X and determining the signatures reduces to evaluating the vector field f at $con(l)$ and determining its sign. Note that the symbol in in the signature of such a discrete state cannot appear. Based on Corollary 1, the signature $s_i(l)$ of an m - order discrete state $l = (l_1, \ldots, l_n)$, $m \geq 1$, is determined by checking what different symbols appear in each of the sets $\{s_i(l') \,|\, l' \in \mathcal{V}(l)\}$, $i = 1, \ldots, n$ [22].

We give here an informal and intuitive description of the algorithm from [22] for finding transitions of DS. For every state $l = (l_1, \ldots, l_n) \in L$, a set L' is created, such that $l \times L'$ contains transitions of DS starting from l, in accordance with Definition 2.

In order to easily describe the transitions from a state with signature entries in the set $\{po, ne, ze\}$, we introduce a map from these symbols to numbers: $eval : \{po, ne, ze\} \to \{+1, -1, 0\}$, $eval(po) = +1$, $eval(ne) = -1$, $eval(ze) = 0$. Each direction i, $i = 1, \ldots, n$ is considered separately and a set L_i containing all sub-labels l'_i of states l' in which l transits is constructed. The main idea in finding elements of set L_i is to decide the value of l'_i based only on the value of $s_i(l)$. Roughly speaking, if $s_i(l) \in \{po, ne, ze\}$, (*i.e.,* $f_i(x)$ has a well defined sign everywhere in $con(l)$ according to Definition 5), then $l'_i = l_i + eval(s_i(l))$. In this case, the added transitions correspond to Definition 2 in which the 'if' statement from Eqn. (4) is replaced by 'if and only if'. It is interesting to note here that our algorithm properly deals with situations in which, judged by the signature $s(l)$ of l, transitions to higher order neighbors l' are suggested, while in reality it is possible that $f(x)$ points towards $con(l')$ everywhere on $con(l)$, while the trajectories of CS only become tangent to $con(l')$ everywhere on $con(l)$ and flow to a even higher order neighbor. Each situation of this kind is signaled by a flag, some preliminary sets L_i, $i = 1, \ldots, n$ are constructed and later they are modified in a fixpoint manner.

If $s_i(l) = in$, then by Definition 5, in general there might exist points in $con(l)$ flowing to all neighbors in direction i, and therefore we let l'_i be any of $\{l_i - 1, l_i, l_i + 1\}$. In this case, it is possible that we add transitions in DS that

do not correspond to trajectories of CS, *i.e.,* Eqn. (4) is satisfied in general with 'if'. However, this source of conservativeness is eliminated through refinement as described below.

After finding all sets L_i, since l can have transitions to its neighbors only, set L' is found by intersecting the cartesian product of sets L_i, $i = 1, \ldots, n$ with the set of neighbors of l.

4.3 Refinement

For a given partition $con(L)$ in which all entries $s_i(l)$, $i = 1, \ldots, n$, in the signatures $s(l)$ of all states $l \in L$ are in the set $\{po, ne, ze\}$, $con(Post(I))$ cannot be shrunk by finer partitioning, for any $I \subset L$. Therefore it does not make sense to partition such quotients.

On the contrary, if for a given partition $con(L)$ there exists a state $l \in L$ and a signature entry $s_i(l) = in$, we can show that proper partitioning produces a discrete quotient $\overline{DS} = (\bar{L}, \bar{T})$ that refines $DS = (L, T)$ in the sense of Definition 3. Therefore, "smaller" over-approximations of the reach set can be constructed (guaranteed strictly smaller if (iii)' in Proposition 1 holds). We give here the main ideas that lead to conclusion that Definition 3 is satisfied.

Rectangles of order 0 (vertices) always have well-defined signature entries $s_i(l)$ in all directions $i = 1, \ldots, n$. A rectangle l of order 1 from DS has indefinite signature entry $s_i(l)$ if $con(l)$ intersects the surface defined by $f_i(x) = 0$ in X. Let l_j be the only odd entry in l. Since f is multi-affine and $con(l)$ is parallel with axis Ox_j, the intersection is a point whose coordinates can be easily computed by solving a linear equation with respect to x_j. Let the solution be denoted by $\tilde{x}_j$. By splitting the current partition DS with respect to the hyperplane $x_j = \tilde{x}_j$, we obtain a new partition $\overline{DS}$. In this partition there are three states refining state l from the previous partition. All these states have well defined signature entry of index i, and by applying the transition algorithm described in Section 4.2 to these states, their discrete transitions will exactly correspond to continuous trajectories in direction i.

A finer quotient $\overline{DS}$ of DS can be found by using a refinement algorithm inspired by the above idea and available in [22]. The algorithm computes all possible intersections in X between all surfaces $f_i = 0$, $i = 1, \ldots, n$ and all $con(l)$, where l is a state of order 1 in DS. Rectangles with order greater than 1 are not split if they have an indefinite signature on a certain direction and all their neighbors of order 1 have well defined signatures on the same direction. From the tests we performed, we observed that if X contains no common points of any two surfaces $f_i = 0$ and $f_j = 0$, $i, j = 1, \ldots, n$, $i \neq j$, then, after a finite number of iterations, the refinement algorithm will not produce new points. In this case, all surfaces $f_i = 0$, $i = 1, \ldots, n$ will eventually have non-empty intersections only with some rectangles of order 0 and of order greater than 1.

4.4 Safety Verification Algorithm

We collect all the results in this paper in the form of an iterative algorithm, detailed in [22], for providing a solution to Problem 2. This safety verification

algorithm starts with an initial rectangular partition determined by the sets I and F. A discrete quotient DS is constructed as described in Sections 4.1 and 4.2 and $\text{Post}(I)$ is calculated using standard techniques from graph theory. If $\text{Post}(I) \bigcap F = \emptyset$, then assertion (2) is true, *i.e.,* $con(F)$ cannot be reached by the continuous system initialized in $con(I)$. If $\text{Post}(I) \bigcap F \neq \emptyset$, then refinement is undertaken as described in Section 4.3. The algorithm is stopped if any of the following occurs: the safety property is satisfied, the refinement is finished, a partitioning precision is reached, or a user defined maximum number of iterations is exceeded. Otherwise, the algorithm iterates by using the finer quotient of DS. When the algorithm is stopped and the safety property is not verified, it returns a sub-region $con(S_F)$ of $con(F)$ which is safe for CS if initialized in $con(I)$. If only an over-approximation of the solution to Problem 1 is desired, then the safety verification algorithm can be run with $F = L$ ($con(F) = X$), where the initial partition L is induced by I only.

On the connection between the solutions to Problems 1 and 2, note that, even if the over-approximation of $con(\text{Post}(I))$ is guaranteed to strictly shrink, this does not necessarily imply that the safe sub-region $con(S_F)$ of $con(F)$ strictly grows. It is guaranteed not to shrink, but it might not grow if the refinement is made in a region of X which has empty intersection with $con(F)$ and/or the rectangles which are refined are not contained in a path from I to F in DS.

5 Case Studies

We have developed a user-friendly software package for Reachability Analysis of Multi-Affine Systems (RAMAS) in Matlab [24]. The program takes as inputs the dimension n, the closed rectangle X, the coefficients $c_{i_1,\ldots,i_n}$ of a multi-affine vector field f as in Eqn. (11), and the sets $con(I)$ and $con(F)$ given in terms of unions of open sub-rectangles of arbitrary order in X. According to algorithm described in Section 4.4, it returns either a positive answer if there are no trajectories of the continuous system from $con(I)$ and $con(F)$, or a subset of $con(F)$ which is guaranteed to be safe with respect to $con(I)$. Even though our tries show that the algorithm works even for $n = 10$, in this paper we focus on a planar case ($n = 2$) so we can show illustrative pictures.

We first consider a nonlinear multi-affine vector field (Case Study 1). We then focus on a linear systems (*i.e.,* $\dot{x} = Ax$) (Case Study 2), which is of course a particular case of multi-affine systems. The qualitative phase portraits for such planar linear systems are known, and reachability properties are almost intuitive. Applying our method to such systems gives us some idea on the conservativeness of our approach, as detailed in [22].

Case Study 1 (nonlinear multi-affine system). Consider $X = [1.5, 3] \times [0.4, 2]$, $f = (f_1, f_2)$ with $f_1 = 2 - x_1x_2$, and $f_2 = 1 + x_2 - x_1x_2$. The initial set is $con(I) = [1.5, 2.5] \times \{0.4\}$, which can be written as the union of two zero-order open rectangles $\{1.5, 0.4\}$, $\{2.5, 0.4\}$ and one first-order open rectangle $(1.5, 2.5) \times 0.4$. The final set is $con(F) = [1.5, 3] \times [0.8, 1.4]$, which in the initial partition can be seen as the union of 6 zero-order open rectangles, 7 first-order

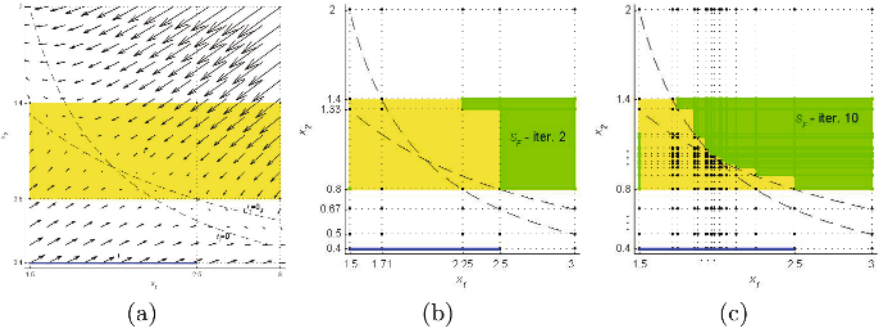

Fig. 2. Case Study 1: (a) Multi-affine vector field f, initial set $con(I)$ (blue - almost black on black and white printers), final set $con(F)$ (yellow - light grey), and initial partition induced by initial and final sets. (b,c) Iterations 2 and 10 from safety verification algorithm. The growing green (dark grey) area represents the safe sub-region $con(S_F)$ of $con(F)$.

open rectangles, and 2 second-order open rectangles. In Fig. 2(a), we plot the vector field f everywhere in X and the two curves $f_1 = 0$ and $f_2 = 0$. Note that the two curves intersect inside $con(F)$, and the refinement procedure will not terminate. At each iteration, the algorithm will produce strictly shrinking over-approximations of Post($con(I)$) in X, which lead to strictly growing safe sub-regions in $con(F)$, as depicted by Fig. 2(b,c).

Case Study 2 (linear system). Consider the rectangular region $X = [-3, 4] \times [-3, 2]$ and the planar linear vector field $f_1 = 0.5x_1 + 1.5x_2$, $f_2 = 1.5x_1 + 0.5x_2$, for which the origin is an unstable node (saddle). The vector field is plotted in Fig. 3(a), together with the initial set $con(I) = [-1, 3] \times \{-2\}$ and the two

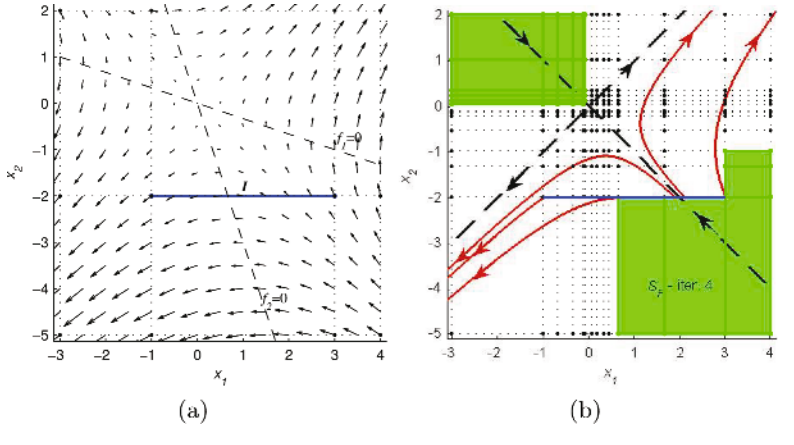

Fig. 3. Case Study 2: (a) Vector field f, lines $f_1 = 0$ and $f_2 = 0$, and initial set (b) Safe region (green - dark grey) obtained in 4 iterations by the reachability algorithm

lines $f_1 = 0$ and $f_2 = 0$, which intersect at the origin. The over-approximation of $Post(con(I))$ calculated in 4 iterations by our method is shown as the white region in Fig. 3(b), together with the eigenvectors and some illustrative trajectories. Note that the refinement does not terminate, but the result does not change significantly with the number of iterations.

6 Conclusion and Future Work

In this paper, we developed a computationally inexpensive method for reachability analysis of multi-affine continuous systems. The method is based on rectangular partitions and iterative constructions of discrete quotients which provide an over-approximation of the reach set of the continuous system, with guaranteed decrease of conservativeness. While falling into the more general framework of [13], where general polynomials are used for partition and polynomial vector fields are allowed, this paper shows that if more structure is allowed, then reachabiltity and safety verification questions can be answered with much less computation. Future work includes development of algorithms to check specifications given in terms of linear temporal logic and and applications to mathematical models found in areas such as biochemistry and control of aircraft and under-water vehicles.

Acknowledgements. This work is partially supported by NSF CAREER 0447721 and NSF 0410514. The second author wishes to thank Luc C.G.J.M. Habets for useful discussions on this topic.

References

1. R. Milner, *Communication and Concurrency.* Prentice Hall, 1989.
2. G. J. Pappas, "Bisimilar linear systems," *Automatica*, vol. 39, no. 12, pp. 2035–2047, 2003.
3. E. Haghverdi, P. Tabuada, and G. Pappas, *Bisimulation relations for dynamical and control systems*, ser. Electronic Notes in Theoretical Computer Science, Blute and e. Peter Selinger, Eds. Elsevier, 2003, vol. 69.
4. T. A. Henzinger, P. W. Kopke, A. Puri, and P. Varaiya, "What is decidable about hybrid automata?" *J. Comput. Syst. Sci.*, vol. 57, pp. 94–124, 1998.
5. R. Alur and D. L. Dill, "A theory of timed automata," *Theoret. Comput. Sci.*, vol. 126, pp. 183–235, 1994.
6. 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 *Lecture Notes in Computer Science.* New York: Springer-Verlag, 1993, vol. 736, pp. 209–229.
7. X. Nicolin, A. Olivero, J. Sifakis, and S. Yovine, "An approach to the description and analysis of hybrid automata," in *Lecture Notes in Computer Science.* New York: Springer-Verlag, 1993, vol. 736, pp. 149–178.
8. A. Puri and P. Varaiya, "Decidability of hybrid systems with rectangular differential inclusions," *Computer Aided Verification*, pp. 95–104, 1994.

9. G. Lafferriere, G. J. Pappas, and S. Sastry, "O-minimal hybrid systems," *Math. Control, Signals, Syst*, vol. 13, no. 1, pp. 1–21, 2000.
10. G. Lafferriere, G. J. Pappas, and S. Yovine, "A new class of decidable hybrid systems," in *Lecture Notes in Computer Science*. New York: Springer-Verlag, 1999, vol. 1569, pp. 137–151.
11. ——, "Reachability computation for linear hybrid systems," in *Proc. 14th IFAC World Congress*, Beijing, P.R.C, July 1999.
12. R. Alur, T. Dang, and F. Ivancic, "Reachability analysis of hybrid systems via predicate abstraction," in *Fifth International Workshop on Hybrid Systems: Computation and Control*, Stanford, CA, 2002.
13. A. Tiwari and G. Khanna, "Series of abstractions for hybrid automata," in *Fifth International Workshop on Hybrid Systems: Computation and Control*, Stanford, CA, 2002.
14. R. Ghosh, A. Tiwari, and C. Tomlin, "Automated symbolic reachability analysis; with application to delta-notch signaling automata," in *Lecture Notes in Computer Science*. New York: Springer-Verlag, 2003, vol. 2623, pp. 233–248.
15. L. Habets and J. van Schuppen, "A control problem for affine dynamical systems on a full-dimensional polytope," *Automatica*, vol. 40, pp. 21–35, 2004.
16. C. Belta and L. Habets, "Constructing decidable hybrid systems with velocity bounds," in *43rd IEEE Conference on Decision and Control*, Paradise Island, Bahamas, 2004.
17. C. Belta, V. Isler, and G. J. Pappas, "Discrete abstractions for robot planning and control in polygonal environments," *IEEE Trans. on Robotics*, vol. 21, no. 5, pp. 864–874, 2005.
18. C. Belta and L. Habets, "Control of a class of nonlinear systems on rectangles," *IEEE Transactions on Automatic Control*, 2005, to appear.
19. C. Belta, "On controlling aircraft and underwater vehicles," in *IEEE International Conference on Robotics and Automation*, New Orleans, LA, 2004.
20. V. Volterra, "Fluctuations in the abundance of a species considered mathematically," *Nature*, vol. 118, pp. 558–560, 1926.
21. A. Lotka, *Elements of physical biology*. New York: Dover Publications, Inc., 1925.
22. M. Kloetzer and C. Belta, "Reachability analysis of multi-affine systems," Boston University, Brookline, MA, Technical report CISE-2005-IR-0070, October 2005. [Online]. Available: http://www.bu.edu/systems/research/publications/2005/2005-IR-0070.pdf
23. P. Tabuada and G. Pappas, "Model checking LTL over controlable linear systems is decidable," in *Hybrid Systems : Computation and Control*, ser. Lecture Notes in Computer Science. Springer Verlag, 2003, vol. 2623.
24. M. Kloetzer and C. Belta, "Reachability analysis of multi-affine systems (ramas)," URL http://iasi.bu.edu/∼software/reach-ma.htm.