SYNTHESIS FOR CONSTRAINED NONLINEAR SYSTEMS USING HYBRIDIZATION AND ROBUST CONTROLLERS ON SIMPLICES

ANTOINE GIRARD AND SAMUEL MARTIN

ABSTRACT. In this paper, we propose an approach to controller synthesis for a class of constrained nonlinear systems. It is based on the use of a hybridization, that is a hybrid abstraction of the nonlinear dynamics. This abstraction is defined on a triangulation of the state-space where on each simplex of the triangulation, the nonlinear dynamics is conservatively approximated by an affine system subject to disturbances. Except for the disturbances, this hybridization can be seen as a piecewise affine hybrid system on simplices for which appealing control synthesis techniques have been developed in the past decade. We extend these techniques to handle systems subject to disturbances by synthesizing and coordinating local robust affine controllers defined on the simplices of the triangulation. We show that the resulting hybrid controller can be used to control successfully the original constrained nonlinear system. Our approach, though conservative, can be fully automated and is computationally tractable. To show its effectiveness in practical applications, we apply our method to control a pendulum mounted on a cart.

1. INTRODUCTION

Abstraction based approaches to control synthesis have become quite popular (see [20] and the references therein). These consist in computing a simple conservative approximation (i.e. the abstraction) of the system dynamics. A controller synthesized using the abstraction can then be used to control the original system. Most of these approaches use discrete abstractions since these allow to leverage controller synthesis techniques developed in the area of discrete event systems. The use of hybrid abstractions, called hybridizations, though common for verification purpose [13, 1, 16, 5], has been much less explored in the context of controller synthesis.

In this paper, we present an approach to control synthesis for a class of constrained nonlinear systems based on the use of a hybridization. The abstraction is defined on a partition of the state-space in simplices (i.e. a triangulation). In each of these simplices, the nonlinear dynamics is approximated by an affine system with disturbances where the disturbances account for the approximation errors. The resulting abstraction can be seen as a piecewise affine hybrid system on simplices subject to disturbances. For such systems, without disturbances, automated control synthesis techniques have been developed by Habets and van Schuppen in a series of papers [11, 12, 10] (see also the related work in [4, 14, 19]). We extend these techniques to handle the systems subject to disturbances by synthesizing local robust affine controllers defined on the simplices of the triangulation. We show that by coordinating these local controllers it is then possible to control successfully the original constrained nonlinear system. A preliminary version of these results was presented in [9]. The technical content in the present version has been improved: the class of considered abstractions is more general (the disturbance is constrained in an arbitrary compact convex set); we synthesize hybrid controllers instead of piecewise smooth controllers, this allows us to prove the correctness

---

of the synthesized controller more rigorously; finally, a more challenging application example is considered.

The paper is organized as follows. In the next section, we formulate the control problem under consideration. In section 3, we introduce the notion of hybridizations and show how these can be computed for nonlinear systems. In section 4, we extend the techniques of [10, 19] for computing local robust affine controllers on simplices. In section 5, we show how to use these local controllers and the hybridization in order to synthesize a controller for the original system. Finally, in section 6, we show the effectiveness of our approach by applying it to control a pendulum mounted on a cart.

2. Problem Formulation

In this paper, we consider a constrained nonlinear system of the form:

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = f(\mathbf{x}(t)) + g(\mathbf{x}(t))\mathbf{u}(t), \\ \mathbf{x}(t) \in X, \ \mathbf{u}(t) \in U \end{cases}$$

where the state domain $X \subseteq \mathbb{R}^n$ is the union of a finite number of compact polytopes and the control domain $U \subseteq \mathbb{R}^m$ is a compact polytope. We shall assume that the maps $f : X \to \mathbb{R}^n$ and $g : X \to \mathbb{R}^{n \times m}$ are of class $\mathcal{C}^2$ and $\mathcal{C}^1$ respectively. Let $X_I \subseteq X$ and $X_T \subseteq X$ be compact polytopes, specifying a set of initial states and a set of target states, respectively.

Definition 2.1. A *hybrid controller* for system Σ is a tuple $\mathcal{C} = (Q, E, \mathcal{D}, \mathcal{G}, \mathcal{H})$, where Q is a finite set of modes; $E \subseteq Q \times Q$ is the set of edges; $\mathcal{D} = \{D_q \subseteq X | \ q \in Q\}$ is the set of domains with D_q a compact set for all $q \in Q$; $\mathcal{G} = \{G_e \subseteq D_q \cap D_{q'} | \ e = (q, q') \in E\}$ is the set of guards with G_e a compact set for all $e \in E$; $\mathcal{H} = \{h_q : D_q \to U | \ q \in Q\}$ is the set of continuous controllers with h_q a Lipschitz continuous map for all $q \in Q$.

The system Σ controlled by $\mathcal{C}$ is a hybrid system, denoted $\Sigma \| \mathcal{C}$ whose evolution is described by executions [15]. An *execution* of $\Sigma \| \mathcal{C}$ is a (finite or infinite) sequence of triples $(\tau_i, q_i, \mathbf{x}_i)_{i=0}^N$ consisting of time intervals $\tau_i = [t_i, t_{i+1}]$ (if N is finite we can have $\tau_N = [t_N, +\infty)$), modes $q_i \in Q$, and differentiable maps $\mathbf{x}_i : \tau_i \to X$ such that $t_0 = 0$, $\mathbf{x}_0(t_0) \in X_I \cap D_{q_0}$; for all $i = 0, \ldots, N$ with $i < +\infty$, for all $t \in [t_i, t_{i+1}]$

$$\mathbf{x}_i(t) \in D_{q_i} \text{ and } \dot{\mathbf{x}}_i(t) = f(\mathbf{x}_i(t)) + g(\mathbf{x}_i(t))h_{q_i}(\mathbf{x}_i(t));$$

and for all $i < N$,

$$(q_i, q_{i+1}) \in E, \text{ and } \mathbf{x}_i(t_{i+1}) = \mathbf{x}_{i+1}(t_{i+1}) \in G_{(q_i, q_{i+1})}.$$

The execution is said *infinite* if N is infinite or $\tau_N = [t_N, +\infty)$, otherwise it is said *finite*; it is said *Zeno* if N is infinite and $\lim_{i \to +\infty} t_i$ is finite. $\Sigma \| \mathcal{C}$ is said to be *non-blocking* if all finite executions can be extended into infinite ones; it is said *non-Zeno* if it does not have Zeno executions. We consider the following synthesis problem:

Problem 2.2. Synthesize a hybrid controller $\mathcal{C}$, such that $X_I \subseteq \bigcup_{q \in Q} D_q$ and the hybrid system $\Sigma \| \mathcal{C}$ is non-blocking, non-Zeno and all its infinite executions $(\tau_i, q_i, \mathbf{x}_i)_{i=0}^N$ satisfy

$$\exists i^* \in \mathbb{N}, \ 0 \le i^* \le N, \ \forall i = i^*, \ldots, N, \ \forall t \in \tau_i, \ \mathbf{x}_i(t) \in X_T.$$

Let us remark that since for all $q \in Q$, $D_q \subseteq X$, all infinite executions of $\Sigma \| \mathcal{C}$ also satisfy for all $i = 0, \ldots, N$, for all $t \in \tau_i$, $\mathbf{x}_i(t) \in X$. Our approach to Problem 2.2 consists of two main ingredients: a hybridization of the nonlinear system Σ and robust continuous controllers defined on simplices. These are presented in the following sections.

Notations. In the following, for two vectors v_1, v_2, $v_1 \cdot v_2$ denotes their scalar product. For a vector v, $\|v\| = \sqrt{v \cdot v}$ is its Euclidean norm. For a matrix A, $\|A\|$ denotes its norm induced by the Euclidean norm for vectors, and $\rho(A)$ denotes its spectral radius.

3. Hybridization

A hybridization is a hybrid abstraction of a continuous dynamical system [1]. Conservativeness of the approximation is ensured by the introduction of disturbances. In this paper, we shall use hybridizations for control synthesis. We first describe the hybridization principle.

Definition 3.1. $\mathcal{S} = \{S_p | \ p \in P\}$ is a *triangulation* of the domain X if the following conditions hold: for all $S_p \in \mathcal{S}$, S_p is a full dimensional simplex of $\mathbb{R}^n$ (i.e. S_p is the convex hull of $n+1$ affinely independent points); for all S_p, $S_{p'} \in \mathcal{S}$, their intersection is the convex hull of their common vertices or empty; $\bigcup_{p \in P} S_p = X$.

We further assume that $\mathcal{S}$ contains a triangulation of X_T, $\mathcal{S}_T \subseteq \mathcal{S}$. The main idea of the hybridization principle consists in approximating the dynamics of Σ locally, in each simplex $S_p \in \mathcal{S}$, by an affine dynamics with disturbances of the form:

$$\begin{cases} \dot{\mathbf{x}}(t) = A_p \mathbf{x}(t) + B_p \mathbf{u}(t) + a_p + \mathbf{w}(t), \\ \mathbf{u}(t) \in U, \ \mathbf{w}(t) \in W_p \end{cases}$$

where the disturbance set W_p is assumed to be a non empty convex compact set; let us define $\Sigma' = \{(S_p, A_p, B_p, a_p, W_p) | \ p \in P\}$. With the exception of the disturbance, the system Σ' can be seen as a piecewise affine hybrid systems on simplices for which control synthesis techniques have been developed in [10]. The disturbance is added to compensate the approximation error and guarantees the conservativeness of the approximation:

Definition 3.2. Σ' is a *hybridization* of system Σ if and only if for all $p \in P$,

(3.1) $$\forall x \in S_p, \ \forall u \in U, \ (f(x) + g(x)u - A_p x - B_p u - a_p) \in W_p.$$

We now propose a method for the computation of a hybridization. We do not discuss the computation of a triangulation $\mathcal{S}$ of the domain X. This is a well studied problem in computational geometry for which efficient algorithms exist, at least in low dimensional spaces (see *e.g.* [17]). In higher dimensional spaces, provided the domain can be partionned in hypercubes, a triangulation of X can be obtained using a simple triangulation of each hypercube as shown in [1]. It is to be noted that the size of the simplices of the triangulation can generally be made arbitrary small.

We now focus on the computation of the affine dynamics in simplex $S_p \in \mathcal{S}$. Let $\{v_{p,0}, \ldots, v_{p,n}\}$ denote the $n+1$ vertices of the simplex. Essentially, the map $f(x)$ is approximated by the affine vector field $A_p x + a_p$ while the map $g(x)$ is approximated by the constant matrix B_p. A_p and a_p can be obtained by interpolation of f at the vertices of the simplex:

$$\forall i \in \{0, \ldots, n\}, \ A_p v_{p,i} + a_p = f(v_{p,i}).$$

Since $v_{p,0}, \ldots, v_{p,n}$ are affinely independent, this condition uniquely determines A_p and a_p which can be computed by solving a set of linear equations. Then, the matrix B_p can be chosen as the barycentric combination of the value of g at the vertices of the simplex:

$$B_p = \frac{1}{n+1} \sum_{i=0}^{n} g(v_{p,i}).$$

It remains to compute the disturbance set W_p such that equation (3.1) holds. For $l \in \{1, \ldots, n\}$, let $f_l(x)$ denote the l-th component of the vector $f(x)$, $g_{l\bullet}(x)$ denote the l-th line l of the matrix $g(x)$, $a_{p,l}$ denote the l-th element of a_p, $A_{p,l\bullet}$ and $B_{p,l\bullet}$ denote the l-th lines of A_p and B_p. The set of disturbances W_p can be chosen as $W_p = [\underline{\mu}_{p,1}, \overline{\mu}_{p,1}] \times \cdots \times [\underline{\mu}_{p,n}, \overline{\mu}_{p,n}]$ where for $l \in \{1, \ldots, n\}$,

$$\underline{\mu}_{p,l} = \min_{x \in S_p, u \in U} (f_l(x) + g_{l\bullet}(x)u - A_{p,l\bullet}x - B_{p,l\bullet}u - a_{p,l})$$

and

$$\overline{\mu}_{p,l} = \max_{x \in S_p, u \in U} (f_l(x) + g_{l\bullet}(x)u - A_{p,l\bullet}x - B_{p,l\bullet}u - a_{p,l}) \,.$$

Then, it is clear that equation (3.1) holds. The following result shows that for the hybridization defined above, the disturbance set W_p can be made arbitrarily small provided the triangulation of the domain is sufficiently fine.

Proposition 3.3. *Let* $\Sigma' = \{(S_p, A_p, B_p, a_p, W_p)| \ p \in P\}$ *be the hybridization defined above. Let* $H(f_l)$ *denote the Hessian matrix of* f_l *and* $J(g_{l\bullet})$ *denote the Jacobian matrix of* $g_{l\bullet}$. *Let* δ_p *denote the diameter of the simplex* S_p: $\delta_p = \max\{\|x - x'\| | \ x, x' \in S_p\}$. *Then,* $W_p \subseteq [-\mu_{p,1}, \mu_{p,1}] \times \cdots \times [-\mu_{p,n}, \mu_{p,n}]$ *where for* $l \in \{1, \ldots, n\}$,

$$\mu_{p,l} = \alpha_{p,l}\delta_p^2 + \beta_{p,l}\delta_p \max_{u \in U} \|u\|$$

with $\alpha_{p,l} = \dfrac{1}{2} \max_{x \in S_p} \rho(H(f_l)(x))$ *and* $\beta_{p,l} = \max_{x \in S_p} \|J(g_{l\bullet})(x)\|.$

Proof. It is clearly sufficient to show that for all $x \in S_p$, for all $l \in \{1, \ldots, n\}$

$$|f_l(x) - A_{p,l\bullet}x - a_{p,l}| \leq \alpha_{p,l}\delta_p^2 \text{ and } \|g_{l\bullet}(x) - B_{p,l\bullet}\| \leq \beta_{p,l}\delta_p.$$

A proof of the first inequality can be found in [21]. The second inequality is derived as follows. From the mean value theorem for vector-valued function we have for all x_1, x_2 in S_p

$$\|g_{l\bullet}(x_1) - g_{l\bullet}(x_2)\| \leq \max_{x \in S_p} \|J(g_{l\bullet})(x)\| \|x_1 - x_2\| \leq \max_{x \in S_p} \|J(g_{l\bullet})(x)\|\delta_p.$$

Then, it follows that for all $x \in S_p$,

$$\|g_{l\bullet}(x) - B_{p,l\bullet}\| \ \leq \ \left\|g_{l\bullet}(x) - \frac{1}{n+1} \sum_{i=0}^{n} g_{l\bullet}(v_{p,i})\right\| \leq \frac{1}{n+1} \sum_{i=0}^{n} \|g_{l\bullet}(x) - g_{l\bullet}(v_{p,i})\|$$

$$\leq \ \frac{1}{n+1} \sum_{i=0}^{n} \max_{x \in S_p} \|J(g_{l\bullet})(x)\|\delta_p = \max_{x \in S_p} \|J(g_{l\bullet})(x)\|\delta_p.$$

$\square$

We can see that the bound on the disturbance depends linearly in the diameter of the simplex. This means that to reduce the bounds by factor $1/2$ it is necessary to consider 2^n times more simplices in the trinagulation. Hopefully, as we shall see on a practical example, it is not always necessary to consider very fine partitions for our approach to be successful.

The rest of the paper is devoted to solving Problem 2.2. Our approach is inspired by the work presented in [10] for piecewise affine hybrid systems on simplices.

4. Robust Controllers on Simplices

We need to extend several techniques developed for the class of affine systems on simplices to the class of systems with disturbances. The results are stated without proofs which are straightforward adaptations of the proofs in [10, 19] and can be found in [9]. Let S be a simplex of $\mathbb{R}^n$, we denote $v_0, \ldots, v_n$ and $F_0, \ldots, F_n$ the vertices and the facets of S with the convention that F_j is the facet opposite to vertex v_j. $m_0, \ldots, m_n$ denote the outward unit normal vectors of the facets of S.

4.1. **Affine systems with disturbances.** We consider the following autonomous affine system with disturbances :

$$(4.1) \qquad \dot{\mathbf{x}}(t) = A\mathbf{x}(t) + a + \mathbf{w}(t), \ \mathbf{x}(t) \in \mathbb{R}^n, \ \mathbf{w}(t) \in W$$

where the disturbance set W is a non empty convex compact set. It is sufficient to assume that the disturbance $\mathbf{w}$ is a continuous map. We say that a trajectory $\mathbf{x}$ of (4.1), starting in S, *exits S* at time $T \geq 0$, if there exists $\varepsilon > 0$ such that

$$\forall t \in [0, T], \ \mathbf{x}(t) \in S \text{ and } \forall t \in (T, T + \varepsilon), \ \mathbf{x}(t) \notin S.$$

We shall say that a facet F_j is *blocked* if

$$\forall x \in F_j, \ m_j \cdot (Ax + a) \leq -\max_{d \in D} m_j \cdot d.$$

The fact that a facet is blocked can be characterized using only the value of the vector field at the vertices:

Proposition 4.1. *The facet F_j is blocked if and only if*

$$\forall i \in \{0, \ldots, n\}, \ i \neq j, \ m_j \cdot (Av_i + a) \leq -\max_{d \in D} m_j \cdot d.$$

The following proposition characterizes the points where the trajectories may exit the simplex.

Proposition 4.2. *If a trajectory $\mathbf{x}$ of (4.1), starting in S, exits S at time T, then $\mathbf{x}(T)$ belongs to a facet of S that is not blocked.*

We now give necessary and sufficient conditions such that all the trajectories of (4.1), starting in S, exit S in finite time.

Proposition 4.3. *All the trajectories of (4.1) starting in S, exit S in finite time if and only if*

$$\exists \xi \in \mathbb{R}^n, \ \forall x \in S, \ \xi \cdot (Ax + a) > -\min_{w \in W} \xi \cdot w.$$

The previous property can again be characterized using only the value of the affine vector field at the vertices of the simplex:

Proposition 4.4. *All the trajectories of (4.1) starting in S, exit S in finite time if and only if*

$$\exists \xi \in \mathbb{R}^n, \ \forall i \in \{0, \ldots, n\}, \ \xi \cdot (Av_i + a) > -\min_{w \in W} \xi \cdot w.$$

4.2. **Robust controller synthesis.** Let us now consider an affine control systems with disturbances on simplices of the form:

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t) + a + \mathbf{w}(t),\ \mathbf{x}(t) \in \mathbb{R}^n,\ \mathbf{u}(t) \in U,\ \mathbf{w}(t) \in W$$

where the control domain $U \subseteq \mathbb{R}^p$ is a convex compact polytope and the disturbance set W is a non empty convex compact set. The disturbance $\mathbf{w}$ is assumed to be continuous. We consider the following control problem:

Problem 4.5. Consider a subset of indices $\mathcal{J} \subseteq \{0, \ldots, n\}$, and the associated subset of facets $\mathcal{F} = \{F_j|,\ j \in \mathcal{J}\}$, design an affine feedback controller $h : \mathbb{R}^n \to \mathbb{R}^m$, $h(x) = Kx + k$, such that for all $x \in S$, $h(x) \in U$ and for the autonomous affine system with disturbances

$$(4.2) \qquad\qquad \dot{\mathbf{x}}(t) = A\mathbf{x}(t) + Bh(\mathbf{x}(t)) + a + \mathbf{w}(t),\ \mathbf{x}(t) \in \mathbb{R}^n,\ \mathbf{w}(t) \in W$$

all facets that are not in $\mathcal{F}$ are blocked and all trajectories of (4.2) starting in S exits S in finite time.

We denote $u_0, \ldots, u_n \in U$ the values of the controller at the vertices of S:

$$u_i = h(v_i) = Kv_i + k,\ i \in \{0, \ldots, n\}.$$

Since $v_0, \ldots, v_n$ are affinely independent, $u_0, \ldots, u_n$ uniquely determine the matrix K and the vector k. Moreover since $u_0, \ldots, u_n \in U$ and for all $x \in S$, $h(x)$ is a convex combination of $u_0, \ldots, u_n$ it follows that $h(x) \in U$, for all $x \in S$. At the vertices of the simplex S, the value of the vector field of (4.2) is given by $Av_i + Bh(v_i) + a = Av_i + Bu_i + a$. In the following we characterize suitable values of $u_0, \ldots, u_n$ with the understanding that these allows the computation of the affine controller h. The following result is a direct consequence of Propositions 4.1 and 4.4.

Proposition 4.6. *For $i \in \{0, \ldots, n\}$, let us consider the following convex polytopes:*

$$U_i = \left\{ u \in U \ \middle|\ m_j \cdot (Av_i + Bu + a) \le -\max_{d \in D} m_j \cdot d,\ \forall j \in \{0, \ldots, n\} \setminus (\mathcal{J} \cup \{i\}) \right\}.$$

An affine feedback controller h solves Problem 4.5 if and only if for all $i \in \{0, \ldots, n\}$, $u_i \in U_i$ and

$$\exists \xi \in \mathbb{R}^n,\ \forall i \in \{0, \ldots, n\},\ \xi \cdot (Av_i + Bu_i + a) > -\min_{w \in W} \xi \cdot w.$$

Let us discuss the computation of input values $u_0, \ldots, u_n$ satisfying the conditions of the previous proposition. We denote by $\overline{U}_i$ the set of vertices of U_i.

Proposition 4.7. *There exists an affine controller h solving Problem 4.5 if and only if there exists $u_0 \in \overline{U}_0, \ldots, u_n \in \overline{U}_n$ such that*

$$\exists \xi \in \mathbb{R}^n,\ \forall i \in \{0, \ldots, n\},\ \xi \cdot (Av_i + Bu_i + a) > -\min_{w \in W} \xi \cdot w.$$

A controller solving Problem 4.5 can therefore be synthesized by computing the vertices of the polytopes $U_0, \ldots, U_n$ and then looking for a suitable combination of these vertices.

5. NONLINEAR CONTROL SYNTHESIS

We now describe a solution to Problem 2.2 based on the results described in the previous sections. Let Σ' be a hybridization of Σ. Our approach is based on the coordination of robust affine controllers defined on the simplices of the triangulation $\mathcal{S}$.

5.1. **Invariance controller.** We start by synthesizing a controller in order to render the set X_T of target states invariant. For all simplices $S_p \in \mathcal{S}_T$ of the triangulation of X_T, we denote by *vertices*(S_p) its set of vertices and by *external-facets*(S_p, X_T) the set of facets of S_p that are included in the boundary of the polytope X_T. For a given facet $F \in$ *external-facets*(S_p, X_T), we denote by m_F its outward unit normal vector and by *vertices*(F) its set of vertices.

Problem 5.1 (Invariance control). Synthesize a Lipschitz continuous controller $h_T : X_T \to U$, such that for all $x_0 \in X_T$, the solution of

(5.1) $$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t)) + g(\mathbf{x}(t))h_T(\mathbf{x}(t)), \ \mathbf{x}(0) = x_0$$

is defined for all $t \in \mathbb{R}^+$.

Let us remark that since h_T is defined only on X_T, this implies that $\mathbf{x}(t) \in X_T$ for all $t \in \mathbb{R}^+$. We will search the controller h_T as a continuous piecewise affine map defined on the triangulation $\mathcal{S}_T$ (i.e. h_T is affine on each simplex of the triangulation). h_T is uniquely determined by its value at the vertices of the simplices of $\mathcal{S}_T$. If all these values belong to U then so does $h_T(x)$, for all $x \in X_T$. Also, it can be shown [8] that h_T, defined this way, is Lispchitz on X_T. Then, from the standard characterization of invariant sets [2] we have the following result:

Lemma 5.2. $h_T : X_T \to U$ solves Problem 5.1 if and only if for all $S_p \in \mathcal{S}_T$, for all $F \in$ external-facets(S_p, X_T), for all $x \in F$, $m_F \cdot (f(x) + g(x)h_T(x)) \leq 0$.

We now characterize the values of h_T at the vertices of the simplices of $\mathcal{S}_T$ that result in controllers solving Problem 5.1:

Proposition 5.3. If for all $S_p \in \mathcal{S}_T$, for all $F \in$ external-facets(S_p, X_T), for all $v \in$ vertices(F)

(5.2) $$m_F \cdot (A_p v + B_p h_T(v) + a_p) \leq - \max_{w \in W_p} m_F \cdot w$$

then h_T solves Problem 5.1.

Proof. Let $S_p \in \mathcal{S}_T$, $F \in$ external-facets(S_p, X_T), and $x \in F$, then

$$\begin{aligned} m_F \cdot (f(x) + g(x)h_T(x)) &= m_F \cdot (A_p x + B_p h_T(x) + a_p) \\ &\quad + m_F \cdot (f(x) + g(x)h_T(x) - A_p x - B_p h_T(x) - a_p). \end{aligned}$$

Since Σ' is a hybridization of Σ, and since $h(x) \in U$, it follows from Definition 3.2 that $(f(x) + g(x)h_T(x) - A_p x - B_p h_T(x) - a_p) \in W_p$. Therefore,

$$m_F \cdot (f(x) + g(x)h_T(x)) \leq m_F \cdot (A_p x + B_p h_T(x) + a_p) + \max_{w \in W_p} m_F \cdot w.$$

Since $A_p x + B_p h_T(x) + a_p$ is an affine map on F, we have by convexity and by (5.2) that

$$m_F \cdot (A_p x + B_p h_T(x) + a_p) + \max_{w \in W_p} m_F \cdot w \leq 0.$$

Then, from Lemma 5.2, we have that the controller h_T solves Problem 5.1. $\qquad \square$

Let us remark that the computation of the controller h_T involves only finding values of h_T, satisfying a set of linear inequalities, at the vertices of the triangulation. This can be done efficiently using linear programming. For simplicity of the presentation, we assumed that we render the whole target set X_T invariant; actually, as far as Problem 2.2 is concerned, it is sufficient to render a subset $X_T' \subseteq X_T$ invariant. This can be done a similar way.

5.2. **Reachability controller.** We now describe how to synthesize a hybrid controller solving Problem 2.2. The proposed controller essentially drives the trajectories of the system through a sequence of simplices ending in X_T. This is done by coordinating robust affine controllers defined in section 4.2.

Let $\mathcal{S}'$ be a subset of the triangulation $\mathcal{S}$ and let $S_p \in \mathcal{S}$, we denote by *common-facets*$(S_p, \mathcal{S}')$ the subset of facets of the simplex S_p that are also facets of a simplex in $\mathcal{S}'$. We denote by *adjacent*$(\mathcal{S}')$ the subset of simplices that are adjacent to a simplex in $\mathcal{S}'$: $S_p \in \text{*adjacent*}(\mathcal{S}')$ if and only if $S_p \in \mathcal{S} \setminus \mathcal{S}'$ and *common-facets*$(S_p, \mathcal{S}') \neq \emptyset$. The synthesis of the hybrid controller $\mathcal{C}$ can be done using the following algorithm:

Algorithm 5.4. Let $Q := \{q_T\}$, $D_{q_T} := X_T$, h_{q_T} solve Problem 5.1, $E := \emptyset$.
Let $k := 0$, $\mathcal{S}_{-1} := \emptyset$, $\mathcal{S}_0 := \mathcal{S}_T$.
While $\mathcal{S}_k \neq \mathcal{S}_{k-1}$:

- $\mathcal{S}_{k+1} := \mathcal{S}_k$.
- For all $S_p \in \text{*adjacent*}(\mathcal{S}_k)$, if there exists an affine feedback controller h_p which solves Problem 4.5 with $(S, \mathcal{F}, A, B, a, W) = (S_p, \text{*common-facets*}(S_p, \mathcal{S}_k), A_p, B_p, a_p, W_p)$, then let $\mathcal{S}_{k+1} := \mathcal{S}_{k+1} \cup \{S_p\}$, $Q := Q \cup \{p\}$, $D_p := S_p$, h_p solution of Problem 4.5 and:
  - For $S_{p'} \in \mathcal{S}_k \setminus \mathcal{S}_T$, if *common-facets*$(S_p, \{S_{p'}\}) \neq \emptyset$ then $E := E \cup \{(p, p')\}$, and $G_{(p,p')} := \text{*common-facets*}(S_p, \{S_{p'}\})$,
  - If *common-facets*$(S_p, \mathcal{S}_T) \neq \emptyset$ then $E := E \cup \{(p, q_T)\}$, $G_{(p,q_T)} := \text{*common-facets*}(S_p, \mathcal{S}_T)$.
- $k := k + 1$.

Return $\mathcal{C} = (Q, E, \mathcal{D}, \mathcal{G}, \mathcal{H})$ where $\mathcal{D} = \{D_q | q \in Q\}$, $\mathcal{G} = \{G_e | e \in E\}$, $\mathcal{H} = \{h_q | q \in Q\}$.

The algorithm essentially applies dynamic programming: $\mathcal{S}_k$ consists of the simplices from which we can drive the system to X_T, using robust affine controllers, by following a sequence of at most k simplices; at each iteration we add to $\mathcal{S}_{k+1}$ the simplices in which we can drive all the trajectories to facets that are shared with some simplices in $\mathcal{S}_k$. Let us remark that the Algorithm 5.4 necessarily stops after $K < |\mathcal{S}|$ iterations and each iteration involves solving a number of linear programs that is at most linear in the number of simplices. Then, the worst-case time complexity of the algorithm is quadratic in the number of simplices. However, it is to be noted that the number of simplices in the triangulation is exponential in the dimension of the state-space. The algorithm is therefore exponential in the dimension of the system. This is expected for an approach based on abstractions defined on a partition of the state-space.

Theorem 5.5. *Let $\mathcal{C}$ be computed by Algorithm 5.4; if $X_I \subseteq \bigcup_{q \in Q} D_q$ then $\mathcal{C}$ solves Problem 2.2.*

Proof. Let us consider a finite execution $(\tau_i, q_i, \mathbf{x}_i)_{i=0}^N$ of $\Sigma \| \mathcal{C}$. If $q_N = q_T$, then since h_{q_T} solves Problem 5.1, $\mathbf{x}_N$ can be extended on $[t_N, +\infty)$. If $q_N \neq q_T$, then $q_N \in P$, let $k \leq K$ be the smallest index such that $S_{q_N} \in \mathcal{S}_k$. Since Σ' is a hybridization of Σ, the continuous dynamics in mode q_N is:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= f(\mathbf{x}(t)) + g(\mathbf{x}(t))h_{q_N}(\mathbf{x}(t)) \\ &= A_{q_N}\mathbf{x}(t) + B_{q_N}h_{q_N}(\mathbf{x}(t)) + a_{q_N} + \mathbf{w}(t), \ \mathbf{w}(t) \in W_{q_N} \end{aligned}$$

for an obvious particular value of the disturbance $\mathbf{w}(t)$. By construction of h_{q_N}, all trajectories of the linear system above exits S_{q_N} in finite time and by Proposition 4.2, $\mathbf{x}_N$ can be extended on $[t_N, t'_{N+1}]$ where $\mathbf{x}_N(t'_{N+1})$ belongs to a facet F_{q_N} of S_{q_N} that is not blocked for the linear system above. By construction of the guards, there exists an edge $e = (q_N, q_{N+1})$, such that $F_{q_N} \subseteq G_e$. Hence, the finite execution can be extended into an execution of the form $(\tau_i, q_i, \mathbf{x}_i)_{i=0}^{N+1}$. Moreover, either $q_{N+1} = q_T$

or we have $S_{q_{N+1}} \in \mathcal{S}_{k-1}$. From the previous discussion, and using a simple induction, we can show that every finite executions of $\Sigma||\mathcal{C}$ can be extended into infinite ones, by taking a finite number of discrete transitions (at most K) before reaching the final mode q_T in which continuous evolution is enabled until time goes to infinity. Hence, $\Sigma||\mathcal{C}$ is non-blocking. Since all executions have only a finite number of discrete transitions, $\Sigma||\mathcal{C}$ is necessarily non-Zeno. Finally, all executions of $\Sigma||\mathcal{C}$ reach the final mode q_T and stays there forever, since $D_{q_T} = X_T$, Problem 2.2 is solved. □

Let us remark that our approach for solving Problem 2.2 is clearly conservative. Our algorithm may fail to solve the problem even though a suitable controller exists. There are several sources of conservatism. The first one is due to the use of a hybridization. Proposition 3.3 suggests that this can be reduced by using a finer triangulation at the price of an increased computational effort. The other sources of conservatism are inherent to the approach developed in [10]. However, this conservatism allows us to synthesize controllers that are correct by design by a fully automated method which is computationally effective as shown in the following section.

6. Example

In this section, we apply our approach for controlling a pendulum mounted on a cart. This system has been considered in [18] for illustrating controller synthesis using discrete abstractions. The dynamics of the system is described by

$$\begin{cases} \dot{\mathbf{x}}_1(t) &= \mathbf{x}_2(t) \\ \dot{\mathbf{x}}_2(t) &= -\sin(\mathbf{x}_1(t)) - \mathbf{u}(t)\cos(\mathbf{x}_1(t)) \end{cases}, \ \mathbf{u}(t) \in [-\alpha, \alpha]$$

where $\mathbf{x}_1(t)$ is the angle between the pendulum and the downward vertical, $\mathbf{x}_2(t)$ is the angular velocity. The control input $\mathbf{u}(t)$ is the acceleration of the cart and is assumed to be bounded in absolute value by α. The goal is to bring the pendulum from downward vertical position ($\mathbf{x}_1(t) = 0$) to upward vertical position ($\mathbf{x}_1(t) = \pi$) and to keep it there. This can be formulated as Problem 2.2 where $X = [0, 2\pi] \times [-\pi, \pi]$, $X_I = \{(0,0)\}$ and X_T is a neighborhood of $(\pi, 0)$. Let us remark that since $\mathbf{x}_1(t)$ represents an angle then if the trajectory exits the domain X with $\mathbf{x}_1(t) = 2\pi$, it re-enters immediately with $\mathbf{x}_1(t) = 0$ and the same velocity. This is important when defining the adjacency relations between simplices of the triangulation of X.

We use a regular triangulation defined on a partition of X into squares of length π/N where $N \in \mathbb{N}$ is a parameter. Each of these squares is then partitionned into two triangles: this can be done using a line of slope 1 or -1 giving two possible orientations for the simplices. The target set X_T is given by

$$X_T = \{(x_1, x_2) \in X| \ \tfrac{(N-1)\pi}{N} \le x_1 \le \tfrac{(N+1)\pi}{N}, \ \tfrac{-\pi}{N} \le x_2 \le \tfrac{\pi}{N}, \ \tfrac{(N-1)\pi}{N} \le x_1 + x_2 \le \tfrac{(N+1)\pi}{N} \ \}.$$

The orientation of the simplices is left free at the beginning and is determined on the fly by Algorithm 5.4. The hybridization is also computed on the fly. Let us remark that only the second component of the vector field is nonlinear, therefore the disturbance set W_p is of the form $W_p = \{0\} \times [\underline{\mu}_{p,2}, \overline{\mu}_{p,2}]$ where the bounds given in Section 3 can be computed explicitly. When computing the continuous controllers, to choose among the possible values at the vertices of the simplex, we try to minimize the variation of the control map within the simplex. In particular, if a constant controller solves the problem then such a controller will be used.

Algorithm 5.4 was implemented in Matlab, and ran on a standard desktop equipped with CPU Pentium 4 (3.20 GHz). We report in Table 1 performances of the algorithm for varying number of simplices and set of inputs. We can check that the computational costs increases as the constraints on

$N, |\mathcal{S}|$ U	10, 800	20, 3200	30, 7200	40, 12800
$[-2, 2]$	6 - No	176 - Yes	591 - Yes	1263 - Yes
$[-3, 3]$	6 - No	102 - Yes	330 - Yes	631 - Yes
$[-4, 4]$	14 - Yes	90 - Yes	251 - Yes	531 - Yes

TABLE 1.   Run times in seconds and success of the synthesis for varying number of simplices and set of inputs.
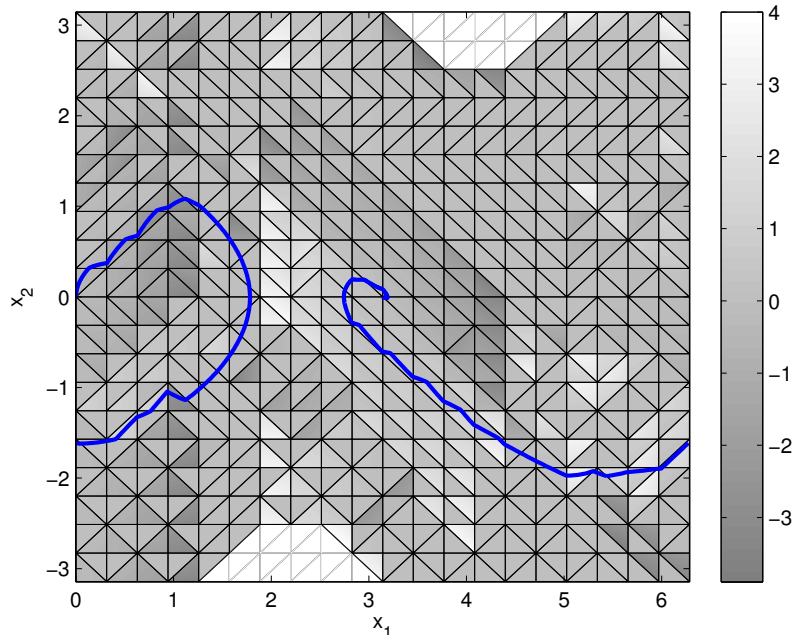

FIGURE 1.   Controller and trajectory of the pendulum on a cart for $\alpha = 4$ and $N = 10$. Levels of gray indicates the value of the control input constrained in $[-4, 4]$.

the inputs become tighter. Also, the complexity with respect to the number of simplices, estimated experimentally on successful cases is roughly in $O(|\mathcal{S}|^{3/2})$ which is polynomial but better than the theoretical worst-case complexity which is quadratic. Compared to the approach presented in [18], where the same example is considered, the approach seems to have quite similar performances: reported computation times are of the same order and the estimated complexity appears to be polynomial in the number of discrete states as well.

In Figure 1, we show the controller obtained for $U = [-4, 4]$ and $N = 10$ (i.e. $|\mathcal{S}| = 800$), as well as the trajectory of the controlled system. The pendulum starts balancing in the anti-clockwise direction to accumulate some energy, then it moves in the clockwise direction until it reaches the target. This nontrivial example shows that our approach can be used for automated controller synthesis for nonlinear systems.

7. Conclusion

In this paper, we presented an algorithmic approach to controller synthesis for constrained nonlinear systems. Our technique is based on two main ingredients, namely a hybridization and robust affine controllers on simplices. Though conservative, our method can be fully automated and we showed that it is effective on an example. Our method should probably be reserved to small-dimensional systems as the number of simplices in the triangulation explodes when the dimension grows.

There are several possible extensions for this work. First, instead of using piecewise affine hybridization defined on a triangulation, one could use piecewise multi-affine hybridization [1] defined on a partition of the state domain in hypercubes as an extension of [3] would allow us to synthesize local controllers on hypercubes. Second, by extending the approach presented in [7, 14], one could solve more complex control problems such as those specified in linear temporal logic (LTL). Finally, the class of nonlinear system we consider could be extended, by considering systems of the form

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)), \ \mathbf{x}(t) \in X, \ \mathbf{u}(t) \in U.$$

Following [6], a piecewise affine hybridization could be computed by triangulating the domain $X \times U$.

References

[1] E. Asarin, T. Dang, and A. Girard, "Hybridization methods for the analysis of non-linear systems," *Acta Informatica*, vol. 43, no. 7, pp. 451–476, 2007.

[2] J.-P. Aubin, *Viability Theory*. Birkhaüser, 1991.

[3] C. Belta and L. Habets, "Controlling a class of nonlinear systems on rectangles," *IEEE Trans. on Automatic Control*, vol. 51, no. 11, pp. 1749–1759, 2006.

[4] C. Belta, V. Isler, and G. Pappas, "Discrete abstractions for robot motion planning and control in polygonal environments," *IEEE Trans. on Robotics*, vol. 21, no. 5, pp. 864–874, 2005.

[5] T. Dang, O. Maler, and R. Testylier, "Accurate hybridization of nonlinear systems," in *Hybrid Systems: Computation and Control*, 2010, pp. 11–20.

[6] J.-G. Dumas and A. Rondepierre, "A hybrid system approach to nonlinear optimal control problems," IMAG, Tech. Rep., 2005. [Online]. Available: http://arxiv.org/abs/math/0502172

[7] G. Fainekos, H. Kress-Gazit, and G. Pappas, "Temporal logic motion planning for mobile robots," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2005, pp. 2020–2025.

[8] A. Girard, "Analyse algorithmique des systèmes hybrides," Ph.D. dissertation, Institut National Polytechnique de Grenoble, 2004.

[9] A. Girard and S. Martin, "Motion planning for nonlinear systems using hybridizations and robust controllers on simplices," in *IEEE Conf. on Decision and Control*, 2008, pp. 239–244.

[10] L. Habets, P. Collins, and J. van Schuppen, "Reachability and control synthesis for piecewise-affine hybrid systems on simplices," *IEEE Trans. on Automatic Control*, vol. 51, no. 6, pp. 938–948, 2006.

[11] L. Habets and J. van Schuppen, "Control of piecewise-linear hybrid systems on simplices and rectangles," in *Hybrid Systems: Computation and Control*, ser. LNCS, vol. 2034. Springer, 2001, pp. 261–274.

[12] ——, "A control problem for affine dynamical systems on a full-dimensional polytope," *Automatica*, vol. 40, pp. 21–35, 2004.

[13] T. Henzinger, P.-H. Ho, and H. Wong-Toi, "Algorithmic analysis of nonlinear hybrid systems," *IEEE Trans. on Automatic Control*, vol. 43, no. 4, pp. 540–554, 1998.

[14] M. Kloetzer and C. Belta, "A fully automated framework for control of linear systems from LTL specifications," in *Hybrid Systems: Computation and Control*, ser. LNCS, vol. 3927. Springer, 2006, pp. 333–347.

[15] J. Lygeros, K. H. Johansson, S. N. Simic, J. Zhang, and S. S. Sastry, "Dynamical properties of hybrid automata," *IEEE Trans. on Automatic Control*, vol. 48, no. 1, pp. 2–17, 2003.

[16] N. Ramdani, N. Meslem, and Y. Candau, "Reachability of uncertain nonlinear systems using a nonlinear hybridization," in *Hybrid Systems: Computation and Control*, ser. LNCS, vol. 4981. Springer, 2008, pp. 415–428.

[17] J. O'Rourke, *Computational geometry in C*. Cambridge Tracts in Theoretical Computer Science, 1998.

[18] G. Reissig, "Computing abstractions of nonlinear systems," *IEEE Trans. on Automatic Control*, 2011, to appear.

[19] B. Roszak and M. Broucke, "Necessary and sufficient conditions for reachability on a simplex," *Automatica*, vol. 42, no. 11, pp. 1913–1918, 2006.

[20] P. Tabuada, *Verification and Control of Hybrid Systems - A Symbolic Approach.*   Springer, 2009.

[21] S. Waldron, "The error in linear interpolation at the vertices of a simplex," *SIAM J. Numer. Anal.*, vol. 35, no. 3, pp. 1191–1200, 1998.

LABORATOIRE JEAN KUNTZMANN, UNIVERSITÉ DE GRENOBLE, B.P. 53, 38041 GRENOBLE, FRANCE

E-mail address: `Antoine.Girard@imag.fr`

LABORATOIRE JEAN KUNTZMANN, UNIVERSITÉ DE GRENOBLE, B.P. 53, 38041 GRENOBLE, FRANCE

E-mail address: `Samuel.Martin@imag.fr`