Automatica 98 (2018) 323–330

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

On the construction of safe controllable regions for affine systems with applications to robotics☆

Mohamed K. Helwa *,[1], Angela P. Schoellig

Dynamic Systems Lab, Institute for Aerospace Studies, University of Toronto, Canada

ARTICLE INFO

Article history:
Received 11 September 2016
Received in revised form 15 February 2018
Accepted 4 August 2018
Available online 10 October 2018

Keywords:
Controllability
Safety constraints
Invariance problems
Robotics
Quadrotors
Obstacle avoidance
Reference feasibility

ABSTRACT

This paper studies the problem of constructing in-block controllable (IBC) regions for affine systems. That is, we are concerned with constructing regions in the state space of affine systems such that all the states in the interior of the region are mutually accessible within the region's interior by applying uniformly bounded inputs. We first show that existing results for checking in-block controllability on given polytopic regions cannot be easily extended to address the question of constructing IBC regions. We then explore the geometry of the problem to provide a computationally efficient algorithm for constructing IBC regions. We also prove the soundness of the algorithm. We then use the proposed algorithm to construct safe speed profiles for robotic systems. As a case study, we present several experimental results on unmanned aerial vehicles (UAVs) to verify the effectiveness of the proposed algorithm; these results include using the proposed algorithm for real-time collision avoidance for UAVs.

1. Introduction

With the increasing desire for building the next generation of engineering systems that can safely interact with their environment and possibly non-professional humans (e.g., self-driving cars or assistive robots), there is an urgent need for developing controller design methods that respect all given safety constraints of the systems even in the transient period. Hence, we set our goal to provide the mathematical foundations for controller design under safety constraints. Although safety constraints can be accounted for using optimal/predictive control strategies (Aswani, Gonzalez, Sastry, & Tomlin, 2013; Qin & Badgwell, 2003), there are many fundamental questions in the area of controller design under safety constraints that still require further studies. For instance, consider a wheeled robot moving on a bounded table, with additional limits on the robot's speed. Using Kalman's controllability notion, we cannot even answer the simple question whether the robot can reach, starting from any initial position and speed, any final position and speed while respecting the safety constraints and using uniformly bounded input force? This shows the urgent need for finding checkable conditions for controllability under safety constraints.

Hence, we recently introduced the study of in-block controllability (IBC), which formalizes controllability under given safety state constraints (Helwa & Caines, 2014a, 2017). The notion of IBC can, however, be motivated from several different perspectives. In Helwa and Caines (2014c, 2015a), we showed that if one constructs a special partition (cover) of the state space of piecewise affine (PWA) hybrid systems (nonlinear systems) such that each region of the partition (cover) satisfies the IBC property, then one can systematically study controllability and build hierarchical structures for the PWA hybrid systems (nonlinear systems). We note that similar to nonlinear systems, controllability of PWA hybrid systems is a challenging open problem to date (Thuan & Camlibel, 2014). Also, building hierarchical structures of dynamical systems allows us to design controllers that achieve temporal logic statements at the higher levels of the hierarchy, and then to systematically realize these high-level control decisions at the lower levels. Moreover, the IBC notion is useful in the context of optimal control problems. In particular, the IBC conditions ensure that all the optimal accessibility problems within given safety constraints are feasible. Furthermore, in this paper we use the IBC results to build safe speed profiles for robotic systems. We then utilize these profiles to achieve obstacle avoidance and to determine the feasibility of given reference trajectories for the robots.

☆ Supported by NSERC grant RGPIN-2014-04634, OCE/SOSCIP TalentEdge Project 27901, and Ontario Ministry of Research, Innovation & Science Early Researcher Award. The material in this paper was partially presented at the 55th IEEE Conference on Decision and Control, December 12–14, 2016, Las Vegas, NV, USA. This paper was recommended for publication in revised form by Associate Editor Ricardo Sanfelice under the direction of Editor Daniel Liberzon.

* Corresponding author.
E-mail addresses: mohamed.helwa@robotics.utias.utoronto.ca (M.K. Helwa), schoellig@utias.utoronto.ca (A.P. Schoellig).
[1] M.K. Helwa is also with the Electrical Power and Machines Department, Cairo University, Egypt.

https://doi.org/10.1016/j.automatica.2018.09.019
0005-1098/

The notion of IBC was utilized to build hierarchical structures of finite state machines, nonlinear systems on closed sets, and automata in Caines and Wei (1995, 1998), and Hubbard and Caines (2002), respectively. However, these papers do not study conditions for when the IBC property holds. In Helwa and Caines (2014a, 2017), three necessary and sufficient conditions were provided for IBC of affine systems on given polytopes. The conditions require solving linear programming (LP) problems at the vertices of the given polytope. In Helwa (2015), the IBC conditions were extended to controlled switched linear systems, while in Helwa and Caines (2014b), the notion of IBC was relaxed to the case where one can distinguish between soft and hard constraints. In Brammer (1972) and Sontag (1984), controllability of linear systems under input constraints was studied, while in Heemels and Camlibel (2007), controllability of continuous-time linear systems under state and/or input constraints was studied under the assumption that the system transfer matrix is right invertible.

In many practical scenarios, however, it may happen that the given affine system is not IBC with respect to (w.r.t.) the given polytope, representing the intersection of the given safety constraints. Hence, it would be important from a practical perspective to find the largest IBC region inside the given region, representing the largest safe region within which we can fully control our system. Also, constructing IBC regions is an essential problem for building the partitions/covers in Helwa and Caines (2014c, 2015a), so that one can make use of the hierarchical control results of these papers.

In this paper, we first show the difficulties of directly using the available results for checking IBC of affine systems on given polytopes to construct IBC regions. Second, we provide a computationally efficient algorithm for constructing IBC polytopes, and prove its soundness. Third, we show how our proposed algorithm can be useful for constructing safe speed profiles for robotic systems. That is, we construct for each position of the robot a corresponding safe speed range. The proposed safe speed profiles are useful for robot speed scheduling algorithms (Kant & Zucker, 1986; Ostafew, Schoellig, Barfoot, & Collier, 2014). If the speed scheduling algorithms limit the selected speeds to our proposed safe speed profiles, then safety of the robot can be always achieved on the given constrained position space by applying a feasible input within the actuation limits. We also show how the proposed safe speed profiles can be used to achieve static/dynamic obstacle avoidance. Moreover, our proposed algorithm guarantees full controllability of the robots on the constructed position–speed regions. Hence, in planning reference trajectories, it would be important to select reference points inside the proposed safe position–speed regions to ensure that they are reachable within the given state constraints and under the actuation limits. Finally, we verify our proposed results through several experimental results on unmanned aerial vehicles (UAVs). Compared to the brief version (Helwa & Schoellig, 2016), we hereby include complete proofs, additional discussions and remarks, and experimental results on UAVs.

Notation: Let $K \subset \mathbb{R}^n$ be a set. The closure of K is denoted by $\overline{K}$, the interior by K°, and the boundary by ∂K. For vectors $x,\ y \in \mathbb{R}^n$, $x \cdot y$ denotes the inner product of the two vectors. The notation $\|x\|$ denotes the Euclidean norm of x. The notation co $\{v_1, v_2, \ldots\}$ denotes the convex hull of a set of points $v_i \in \mathbb{R}^n$.

2. Related work

Compared to the well-known controlled invariance problem (Blanchini, 1999; Dorea & Hennet, 1999), which requires that all the state trajectories initiated in a set to remain in the set for all future time, IBC has the additional requirement of achieving mutual accessibility. Also, unlike the invariant sets, we guarantee that any state in the IBC set is reachable from any other state in the IBC set within the set itself, and consequently, any state in the IBC set can be selected as a point in a feasible reference trajectory for the system. In the literature, several algorithms have been provided for constructing controlled invariant sets. These algorithms can be classified into two main categories (Blanchini, 1999): *(i)* iterative algorithms for finding the largest invariant polytopic sets in given polytopes; these algorithms typically end up with polytopes with high complexity (Athanasopoulos, Bitsoris, & Lazar, 2014; Blanchini, 1999), and *(ii)* eigenstructure analysis algorithms leading to invariant polytopes with low complexity (Blanchini, 1999). Nevertheless, we emphasize that these algorithms cannot be used for building IBC regions, which are different from the invariant regions. Our proposed algorithm is not iterative, and it is based on exploring the geometric structure of the affine system, which has some similarities to the eigenstructure analysis algorithms for constructing invariant sets. Consequently, our algorithm is computationally efficient, and it ends up with polytopic regions with low complexity, which facilitates the construction of feedback laws on the constructed polytopes. For our geometric study of IBC, we utilize some geometric tools used for the study of the reach control problem (RCP) on polytopes; see Broucke (2010), Habets and van Schuppen (2004) and Helwa and Broucke (2013, 2015). Unlike RCP, in IBC, we do not try to force the trajectories of the system to leave the polytope through a prescribed exit facet.

Compared to the feasibility study of Schoellig, Hehn, Lupashin, and D'Andrea (2011), we hereby take the safety position/speed constraints into consideration in determining the feasibility of given references, and not only the robot actuation limits. In Section 6, we verify that the proposed algorithm is computationally efficient by utilizing it to build safe, controllable regions for UAVs online. Then, a control law is provided on the safe region to keep the system inside the region, and hence ensure dynamic obstacle avoidance. Collision avoidance strategies may be classified into: *(i)* motion planning strategies, and *(ii)* reactive control strategies (Rodriguez-Seda et al., 2014). Motion planning strategies calculate a collision-free reference trajectory at initial sampling time based on the estimated position of the obstacles. Fast replanning of collision-free trajectories is needed for dynamic environments (Grzonka, Grisetti, & Burgard, 2012). On the other hand, reactive control strategies continuously calculate updated control inputs online based on obstacles detected. Thus, these strategies are more suitable for fast-moving obstacles (Frew & Sengupta, 2004; Palafox & Spong, 2009; Rodriguez-Seda, Stipanovic, & Spong, 2011; Rodriguez-Seda et al., 2014). Our obstacle avoidance strategy for UAVs is a reactive one, and it has similarities to the strategy in Rodriguez-Seda et al. (2011) for fully-actuated robots and in Rodriguez-Seda et al. (2014) for nonholonomic, two-wheeled, ground vehicles. Unlike most of the obstacle avoidance approaches in the literature, see for instance (Frew & Sengupta, 2004; Palafox & Spong, 2009), our strategy takes the robot dynamics and actuation limits into account, and does not put constraints on the shape/velocity of the moving obstacle.

3. Background

We present some geometric background relevant for the remainder of the paper, see Rockafellar (1970). A set $K \subset \mathbb{R}^n$ is *affine* if $\lambda x + (1-\lambda)y \in K$ for all $x,\ y \in K$ and all $\lambda \in \mathbb{R}$. If the affine set passes through the origin, then it forms a subspace of $\mathbb{R}^n$. For subspaces $\mathcal{A},\ \mathcal{B}, \mathcal{A}+\mathcal{B} := \{a+b\ :\ a \in \mathcal{A},\ b \in \mathcal{B}\}$. The set $\mathcal{A}+\mathcal{B}$ is also a subspace. The *affine hull* of a set K, denoted by aff (K), is the smallest affine set containing K. We mean by a dimension of a set K its affine dimension, which is the dimension of aff (K). A *hyperplane* is an $(n-1)$-dimensional affine set in $\mathbb{R}^n$, dividing $\mathbb{R}^n$ into two open half-spaces. A finite set of vectors $\{x_1, \ldots, x_k\}$ is called *affinely independent* if the unique solution to $\sum_{i=1}^k \alpha_i x_i = 0$ and $\sum_{i=1}^k \alpha_i = 0$ is $\alpha_i = 0$ for all $i = 1, \ldots, k$. Affinely independent vectors do

not all lie in a common hyperplane. An *n-dimensional simplex* is the convex hull of $(n+1)$ affinely independent points in $\mathbb{R}^n$. A simplex is a generalization of a triangle in 2D to arbitrary dimensions. An *n-dimensional polytope* is the convex hull of a finite set of points in $\mathbb{R}^n$ whose affine hull has dimension n. Let $\{v_1, \ldots, v_p\}$ be a set of points in $\mathbb{R}^n$, where $p > n$, and suppose that $\{v_1, \ldots, v_p\}$ contains (at least) $(n+1)$ affinely independent points. Then $X := \text{co}\{v_1, \ldots, v_p\}$ is an n-dimensional polytope. A *face* of X is any intersection of X with a closed half-space such that none of the interior points of X lie on the boundary of the half-space. A *facet* of X is an $(n-1)$-dimensional face of X. A polytope is *simplicial* if all its facets are simplices. We denote the facets of X by $F_1, \ldots, F_r$, and we use h_i to denote the unit normal vector to F_i pointing outside of X. A *triangulation* $\mathbb{T}$ of an n-dimensional polytope X is a finite collection of n-dimensional simplices $S_1, \ldots, S_L$ such that: *(i)* $X = \bigcup_{i=1}^{L} S_i$; *(ii)* for all $i, j \in \{1, \ldots, L\}$ with $i \neq j$, $S_i \cap S_j$ is either empty or a common face of S_i and S_j.

4. In-block controllability

We review IBC. Consider the affine control system:

$$\dot{x}(t) = Ax(t) + Bu(t) + a\,, \qquad x(t) \in \mathbb{R}^n\,, \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$, $a \in \mathbb{R}^n$, $B \in \mathbb{R}^{n\times m}$, and $\text{rank}(B) = m$. Throughout the paper, we assume that the input $u : [0, \infty) \to \mathbb{R}^m$ is measurable and bounded on any compact time interval to ensure the existence and uniqueness of the solutions of (1) (Filippov, 1988). Let $\phi(x_0, t, u)$ be the trajectory of (1) under a control law u, with initial condition x_0 and evaluated at time t. We first review the IBC notion (after (Caines & Wei, 1998)).

Definition 4.1 (*In-Block Controllability (IBC)*)**.** Consider the affine control system (1) defined on an n-dimensional polytope X. We say that (1) is *in-block controllable (IBC)* w.r.t. X if there exists an $M > 0$ such that for all $x,\ y \in X^\circ$, there exist $T \geq 0$ and a control input u defined on $[0, T]$ such that *(i)* $\|u(t)\| \leq M$ and $\phi(x, t, u) \in X^\circ$ for all $t \in [0, T]$, and *(ii)* $\phi(x, T, u) = y$.

In Helwa and Caines (2014a), it was shown that for studying IBC we can always apply a coordinate shift, and assume without loss of generality (w.l.o.g.) that we study

$$\dot{\tilde{x}}(t) = A\tilde{x}(t) + B\tilde{u}(t) \tag{2}$$

on a new polytope $\tilde{X}$ with $0 \in \tilde{X}^\circ$. For notational convenience and w.l.o.g., we will call $\tilde{X}$, $\tilde{x}$, and $\tilde{u}$ just X, x, and u, respectively, in the remainder of the paper. Let $J := \{1, \ldots, r\}$ be the set of indices of the facets of X, and $J(x) := \{j \in J : x \in F_j\}$ be the set of indices of the facets of X in which x is a point. We define the closed, convex *tangent cone* to X at x as $C(x) := \{y \in \mathbb{R}^n : h_j \cdot y \leq 0,\ j \in J(x)\}$, where h_j is the unit normal vector to F_j pointing outside X.

Theorem 4.1 (*Helwa & Caines, 2014a*)**.** *Consider the system (2) defined on an n-dimensional simplicial polytope X satisfying $0 \in X^\circ$. The system (2) is IBC w.r.t. X if and only if (i) (A, B) is controllable; (ii) the so-called invariance conditions of X are solvable (that is, for each vertex $v \in X$, there exists $u \in \mathbb{R}^m$ such that $Av + Bu \in C(v)$); (iii) the so-called backward invariance conditions of X are solvable (that is, for each vertex $v \in X$, there exists $u \in \mathbb{R}^m$ such that $-Av - Bu \in C(v)$).*

In Helwa and Caines (2014a), it was shown that conditions *(i)–(iii)* of Theorem 4.1 are also necessary for IBC on non-simplicial polytopes. For given polytopes, both the invariance and backward invariance conditions can be easily checked by solving a linear programming (LP) problem for each vertex of the polytope. Note that solvability of the invariance and backward invariance conditions at the vertices implies by a simple convexity argument that they are solvable at all boundary points of X.

Remark 4.1. The definition of IBC can be easily tailored to the case when we have both state and input constraints. Suppose $u \in U \subset \mathbb{R}^m$, where U is a polytope having $0 \in U^\circ$. For this case, the system is IBC w.r.t. X if every $x,\ y \in X^\circ$ are mutually accessible within X° using control inputs $u \in U$. Similarly, the definitions of invariance and backward invariance conditions are adapted to restrict u to lie in U. It can be shown that for these tailored definitions, conditions *(i)–(iii)* of Theorem 4.1 remain necessary for IBC. Also, the proof of the sufficiency of conditions *(i)–(iii)* in this case is similar to the one in Section V of Helwa and Caines (2014a) under the mild assumption on U that for any $\bar{x} \in X$ satisfying $A\bar{x} \in \text{Im}\,(B)$, the image of B, there exists a $\bar{u} \in U^\circ$ such that $A\bar{x} + B\bar{u} = 0$.

5. Construction of IBC regions

We study the problem of constructing IBC regions for affine systems. Following Helwa & Caines (2014a), we know that w.l.o.g. the problem of studying IBC of an affine system can be transformed to studying a linear system on a new polytope X having $0 \in X^\circ$. Thus, we consider a linear system (2). Given the necessity of condition *(i)* of Theorem 4.1 for IBC, in our study, we assume w.l.o.g. that (2) is controllable. We then construct around the origin an IBC polytopic region for (2).

Problem 5.1 (*Construction of IBC Polytopes*)**.** Given a controllable linear system (2), construct a polytope X such that $0 \in X^\circ$ and (2) is IBC w.r.t. X.

It can be easily shown that if (2) is IBC w.r.t. the polytope X using uniformly bounded control inputs satisfying $\|u\| \leq M$, then for every $\lambda > 0$, it is also IBC w.r.t. $\lambda X := \{x \in \mathbb{R}^n : x = \lambda y,\ y \in X\}$, a λ-scaled version of X, using control inputs satisfying $\|u\| \leq \lambda M$.

While checking IBC on given polytopes incorporates solving LP problems, building IBC polytopes is considerably more difficult. Theorem 4.1 suggests that we build around the origin simplicial polytopes satisfying both the invariance and backward invariance conditions. Two difficulties are faced here. First, to build a polytope X satisfying the invariance conditions (similar argument holds for the backward invariance conditions), we would need to select the vertices of X, v_i, the unit normal vectors to the facets of X, h_j, and the inputs at the vertices, u_i, such that $h_j \cdot (Av_i + Bu_i) \leq 0$, for all $j \in J(v_i)$. Since h_j, v_i, and u_i are all unknowns, we have a set of bilinear matrix inequalities (BMIs), the solving of which is in general NP-hard (Toker & Ozbay, 1995). Second, even if one constructs X satisfying both the invariance and backward invariance conditions, one still needs to verify that X is simplicial since the proof of the sufficiency of Theorem 4.1 only holds for simplicial polytopes.

We explore the geometry of the problem, and try to provide a computationally efficient algorithm for building IBC polytopes that avoids solving BMIs or using trial-and-error. We initiated this geometric study in Helwa and Caines (2015b) for hypersurface systems with $m = n - 1$, and here we extend the study of Helwa and Caines (2015b) to a more general geometric case. To that end, let $\mathcal{B} := \text{Im}\,(B)$ be the image of B, and define the set of possible equilibria of (2): $\mathcal{O} := \{x \in \mathbb{R}^n : Ax \in \mathcal{B}\}$. At any point in $\mathcal{O}$, the vector field of (2) can vanish by proper selection of the input u. Also, if $x_0 \in \mathbb{R}^n$ is an equilibrium point of (2) under some input, then $x_0 \in \mathcal{O}$ (Broucke, 2010). It can be verified that $\mathcal{O}$ is closed, affine, and its dimension is $m \leq \kappa \leq n$ (Helwa & Broucke, 2013). Note that both $\mathcal{B}$ and $\mathcal{O}$ are properties of the system (2), and, as such, they can be calculated before constructing X. For the geometric case $\mathcal{O} + \mathcal{B} = \mathbb{R}^n$, we provide a computationally efficient algorithm for constructing IBC polytopes. We now show that this condition is more general than the condition $m = n - 1$ considered in Helwa and Caines (2015b). If $m = n - 1$, then the dimension of $\mathcal{O}$ is $n - 1 \leq \kappa \leq n$ (Helwa & Broucke, 2013). If $\kappa = n$, then

$\mathcal{O}+\mathcal{B}=\mathbb{R}^n$ clearly holds. We then show that $\mathcal{O}+\mathcal{B}=\mathbb{R}^n$ holds for the case when $\kappa=n-1$. We claim that $\mathcal{B}$ is not subset of $\mathcal{O}$. Otherwise, we have $Ax+Bu\in\mathcal{B}\subset\mathcal{O}$ for all $x\in\mathcal{O}$, and so $\mathcal{O}$ is an invariant set under any selection of the input u, which contradicts controllability of (2). If $\mathcal{B}$ is not subset of $\mathcal{O}$, then we can identify a non-zero vector $b\in\mathcal{B}$ such that $b\notin\mathcal{O}$. Since $\kappa=n-1$, then clearly $\mathcal{O}+\mathcal{B}=\mathbb{R}^n$. On the other hand, for the following linear system, $\mathcal{O}+\mathcal{B}=\mathbb{R}^n$ holds, while $m<n-1$:

$$\dot{x}(t)=\begin{bmatrix}0&0&0&0\\0&0&0&0\\1&0&1&1\\0&1&0&1\end{bmatrix}x(t)+\begin{bmatrix}1&0\\0&1\\0&0\\0&0\end{bmatrix}u(t). \tag{3}$$

This shows that the geometric case considered in this paper is more general than the one studied in Helwa and Caines (2015b). Since the dimension of $\mathcal{B}$ is m and the dimension of $\mathcal{O}$ is $m\leq\kappa\leq n$ (Helwa & Broucke, 2013), $\mathcal{O}+\mathcal{B}=\mathbb{R}^n$ may be achieved for systems having $m\geq\frac{n}{2}$ as in (3). We start by reviewing a geometric result.

Lemma 5.1 (*Helwa & Caines, 2015b*)**.** *Consider the system (2). For any polytope X, if $v\in\mathcal{O}$ is a vertex of X or if $\mathcal{B}\cap C^\circ(v)\neq\emptyset$ at a vertex v of X, where $C^\circ(v)$ denotes the interior of $C(v)$, then the invariance and backward invariance conditions of X are solvable at v.*

Since $\mathcal{B}$ and $\mathcal{O}$ are properties of the linear system and can be calculated before constructing the polytope X, Lemma 5.1 suggests that we can construct X such that the vertices of X lie on $\mathcal{O}$, or the subspace $\mathcal{B}$ dips into the interior of the tangent cones to X at the vertices. This ensures that both the invariance and backward invariance conditions are solvable. However, there is still the difficulty that the proof of the sufficiency of Theorem 4.1 was carried out in Helwa and Caines (2014a) only for simplicial polytopes, and, consequently, Theorem 4.1 may not apply. Hence, we present the following main result.

Theorem 5.2. *Consider a controllable linear system (2) defined on an n-dimensional polytope X satisfying $0\in X^\circ$. If for each vertex v of X, either $v\in\mathcal{O}$ or $\mathcal{B}\cap C^\circ(v)\neq\emptyset$, then the system (2) is IBC w.r.t. X.*

Proof. By assumption and from Lemma 5.1, both the invariance and backward invariance conditions are solvable at the vertices of X. Although the three conditions of Theorem 4.1 hold, X in our case is not necessarily simplicial, and consequently we cannot exactly follow the same sufficiency proof as in Helwa and Caines (2014a) for Theorem 4.1. Indeed, the proof of Theorem 4.1 is divided into three parts. In the first part, the invariance conditions are used to construct a continuous piecewise linear (PWL) feedback, and under the assumption that X is simplicial, it is proved that all the trajectories initiated in X° eventually tend to $\mathcal{O}$ through X°, and reach close to $\mathcal{O}$ in finite time. In the second part, controllability of (A,B) is used to construct a piecewise continuous control input that makes the trajectories initiated nearby $\mathcal{O}$ slide along $\mathcal{O}$ inside X° towards $0\in X^\circ$ in finite time. Third, using the backward invariance conditions and a similar argument to the first two parts, it is shown that one can steer the backward dynamical system $\dot{x}=-Ax-Bu$ from any state in X° to the origin in finite time within X° using uniformly bounded inputs. Equivalently, one can steer (2) from the origin to any final state in X° in finite time within X° using uniformly bounded inputs. One can see that the assumption that X is simplicial is used in Helwa and Caines (2014a) only in the first part of the proof to show that all trajectories initiated in X° tend to $\mathcal{O}$. As a result, our task is reduced to proving this part in our case for any polytope, not necessarily simplicial. The details are in the Appendix.

We now provide under the condition $\mathcal{O}+\mathcal{B}=\mathbb{R}^n$ a computationally efficient algorithm, Algorithm 1, for constructing a polytope X such that $0\in X^\circ$ and the vertices of X satisfy $v\in\mathcal{O}$ or $\mathcal{B}\cap C^\circ(v)\neq\emptyset$, which implies from Theorem 5.2 that the given system is IBC w.r.t. X.

Algorithm 1 Construction of IBC polytopes

Given: A controllable linear system (2) satisfying $\mathcal{O}+\mathcal{B}=\mathbb{R}^n$; Suppose $\mathcal{B}=\text{sp}\{b_1,\cdots,b_m\}$, and $\{o_{m+1},\cdots,o_n\}$ are such that $o_k\in\mathcal{O}$ for all $k=m+1,\cdots,n$ and $\mathbb{R}^n=\text{sp}\{b_1,\cdots,b_m,o_{m+1},\cdots,o_n\}$.
Objective: Construct an n-dimensional polytope X such that $0\in X^\circ$ and the system (2) is IBC w.r.t. X.
Steps:

(1) Construct an initial n-dimensional polytope P such that $0\in P^\circ$, and let $\{v_1,\cdots,v_p\}$ denote the vertices of P.

(2) Let $T=[b_1\ \cdots\ b_m\ o_{m+1}\ \cdots\ o_n]$ and $T_{\mathcal{O}}=[0\ \cdots 0\ o_{m+1}\ \cdots\ o_n]$. For v_i, $i=1,\cdots,p$, calculate $\bar{o}_i=T_{\mathcal{O}}T^{-1}v_i$.

(3) Select $\alpha>1$, and define $\tilde{o}_i:=\alpha\bar{o}_i$ for $i=1,\cdots,p$.

(4) Define $X:=\text{co}\{v_1,\cdots,v_p,\tilde{o}_1,\cdots,\tilde{o}_p\}$.

Theorem 5.3. *Consider a controllable linear system (2) satisfying $\mathcal{O}+\mathcal{B}=\mathbb{R}^n$. Then, Algorithm 1 always terminates successfully, and (2) is IBC w.r.t. the polytope X.*

Proof. Since $\mathcal{O}+\mathcal{B}=\mathbb{R}^n$, one can always identify $o_{m+1},\ldots,o_n$ such that $o_k\in\mathcal{O}$ for all $k=m+1,\ldots,n$, and $\mathbb{R}^n=\text{sp}\{b_1,\ldots,b_m,o_{m+1},\ldots,o_n\}$. Since T has linearly independent columns, it is invertible. Hence, one can always calculate $\bar{o}_i$, $\tilde{o}_i$, and then construct X. By construction, $0\in P^\circ\subset X^\circ$. We now show that (2) is IBC w.r.t. X. To that end, we prove that the vertices of X satisfy $v\in\mathcal{O}$ or $\mathcal{B}\cap C^\circ(v)\neq\emptyset$. Notice that the vertices of X are subset of $\{v_1,\ldots,v_p,\tilde{o}_1,\ldots,\tilde{o}_p\}$. Let $c_i=(c_{i1},c_{i2},\ldots,c_{in}):=T^{-1}v_i$. It is straightforward to show $v_i=\sum_{j=1}^m c_{ij}b_j+\sum_{j=m+1}^n c_{ij}o_j$, $\sum_{j=1}^m c_{ij}b_j=:b_{v_i}\in\mathcal{B}$, and $\sum_{j=m+1}^n c_{ij}o_j\in\mathcal{O}$. From step 2, $\bar{o}_i=T_{\mathcal{O}}c_i=\sum_{j=m+1}^n c_{ij}o_j\in\mathcal{O}$. Thus, we have $v_i=b_{v_i}+\bar{o}_i$. Since $\mathcal{O}$ is affine and $0\in\mathcal{O}$, $\tilde{o}_i:=\alpha\bar{o}_i\in\mathcal{O}$. We then study the vertices of X in $\{v_1,\ldots,v_p\}$. Notice that $\bar{o}_i\in\text{co}\{\tilde{o}_i,0\}$, and if $\bar{o}_i\neq 0$, then $\tilde{o}_i\neq\bar{o}_i$. Since $\tilde{o}_i\in X$ by construction and $0\in X^\circ$, then $\bar{o}_i\in X^\circ$. If v_i, $i\in\{1,\ldots,p\}$, is a vertex of X and $v_i\notin\mathcal{O}$, then $v_i-b_{v_i}=\bar{o}_i\in X^\circ$ implies that $-b_{v_i}\in\mathcal{B}$ dips into the interior of the tangent cone to X at v_i, i.e., $-b_{v_i}\in\mathcal{B}\cap C^\circ(v_i)\neq\emptyset$. From Theorem 5.2, (2) is IBC w.r.t. X.

Remark 5.1. In step 2 of Algorithm 1, T^{-1} should be calculated only once. Algorithm 1 is not iterative, and it represents a significant reduction of computational complexity compared to the original formulation of the problem that requires solving BMIs or using trial-and-error.

Remark 5.2. As discussed before, for any $\lambda>0$, (2) is also IBC w.r.t. λX using λ-scaled inputs of the ones used to solve mutual accessibility problems on X°. This may be useful in two ways. First, if it is required to keep the system states within given, hard safety constraints that form a region X_c around the origin, then one can first use Algorithm 1 to construct an IBC polytope X satisfying $0\in X^\circ$, and then one can simply scale X such that $\lambda X\subset X_c$. Second, for the case of input constraints ($u\in U\subset\mathbb{R}^m$, where $0\in U^\circ$), one can similarly scale X such that on λX, $\lambda<1$, the IBC property is achieved using $u\in U$.

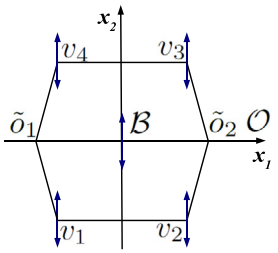

Fig. 1. The constructed IBC polytope X in Example 5.1.

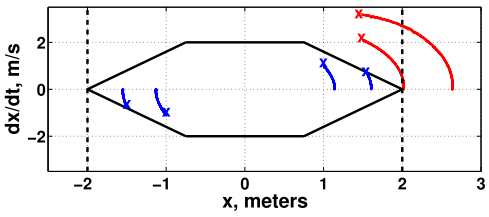

Fig. 2. The IBC region in the x-direction for the position safety constraints $-2 \leq x \leq 2$ and under the actuation limits $|\theta_d| \leq 0.32$ rad, and samples of actual trajectories starting at different initial states (marked with a cross) inside and outside the IBC region.

Remark 5.3. Finding the largest, scaled-version of X in X_c can be achieved by solving the LP problem: $\max \lambda$ such that $\lambda v_i \in X_c$ and $\lambda \tilde{o}_i \in X_c$, for each $i \in \{1, \ldots, p\}$. If u is restricted to lie in U, where $0 \in U^\circ$, then the LP becomes $\max \lambda$ such that $\lambda v_i \in X_c$, $\lambda \tilde{o}_i \in X_c$, $\lambda u_{v_i} \in U$, $\lambda u_{\tilde{o}_i} \in U$, $\lambda u_{b,v_i} \in U$, and $\lambda u_{b,\tilde{o}_i} \in U$, for each $i \in \{1, \ldots, p\}$, where u_{v_i}, $u_{\tilde{o}_i}$ (u_{b,v_i}, $u_{b,\tilde{o}_i}$) are selected inputs at v_i, $\tilde{o}_i$, respectively, satisfying the invariance (backward invariance) conditions of X. Finding the largest IBC polytope inside a safe set is more complex, and may require more computationally complex algorithms.

Remark 5.4. The proof of Theorem 5.2 in the Appendix provides a systematic method for constructing a continuous PWL feedback $u_p(x)$ satisfying the invariance property on the constructed IBC polytope X. Moreover, on X, one can follow the systematic procedure discussed in Remark 5.1 of Helwa and Caines (2017) to construct bounded control inputs satisfying the constrained mutual accessibility between any two given states $x_0,\ x_f \in X^\circ$. Note that, however, this solution satisfying the constrained mutual accessibility is neither unique nor optimal. For the optimal solution, one can utilize optimal/predictive control strategies to find the optimal trajectory connecting two pair of states $x_0,\ x_f \in X^\circ$ within X°, which is a feasible problem since X satisfies the IBC property.

Example 5.1. Consider the double integrator $\dot{x}_1 = x_2, \dot{x}_2 = u$. The system is controllable. We have $\mathcal{O} = \{x \in \mathbb{R}^2 : x_2 = 0\}$, the x_1-axis, and $\mathcal{B} = \text{sp}\{(0, 1)\}$, the x_2-axis. Hence, $\mathcal{O}+\mathcal{B} = \mathbb{R}^2$. We follow the steps of Algorithm 1: *(1)* We construct $P = \text{co}\{v_1, \ldots, v_4\}$, where $v_1 = (-0.8, -1)$, $v_2 = (0.8, -1)$, $v_3 = (0.8, 1)$, and $v_4 = (-0.8, 1)$. From Theorem 4.1, the system is not IBC w.r.t. P; *(2)* we have $b_1 = (0, 1)$, $o_2 = (1, 0)$, and we calculate $\bar{o}_1 = \bar{o}_4 = (-0.8, 0)$ and $\bar{o}_2 = \bar{o}_3 = (0.8, 0)$; *(3)* we select $\alpha = 1.25$, and so $\tilde{o}_1 = \tilde{o}_4 = (-1, 0)$ and $\tilde{o}_2 = \tilde{o}_3 = (1, 0)$; *(4)* the system is IBC w.r.t. $X = \text{co}\{v_1, \ldots, v_4, \tilde{o}_1, \tilde{o}_2\}$ shown in Fig. 1. ◁

6. Applications to unmanned aerial vehicles

We utilize our proposed algorithm to construct safe, controllable position–speed regions for an important class of UAVs, namely quadrotors, and then show using experimental results on a Parrot AR.Drone 2.0 platform how these regions can be useful for the safe control of UAVs in confined spaces and under actuation limits. The quadrotor has six degrees of freedom: the translational position (x, y, z), measured in the inertial coordinate frame O, and the vehicle Euler angles (ϕ, θ, ψ), rotating the inertial frame into the body-fixed frame $\mathcal{V}$, where ϕ is the roll angle, θ is the pitch angle, and ψ is the yaw angle. Let $s := (x, y, z)$. The translational dynamics of the quadrotor are $\ddot{s} = R_{ZYX}(\psi, \theta, \phi)\, \bar{f} - \bar{g}$, where $\bar{f} = (0, 0, f)$, f is the sum of the four rotor forces normalized by the vehicle mass, $\bar{g} = (0, 0, g)$, $g = 9.8$ m/s^2, and $R_{ZYX}(\psi, \theta, \phi)$ is the rotation matrix from $\mathcal{V}$ to O; see Schoellig et al. (2011). The Parrot AR.Drone 2.0 platform has an onboard controller that takes four inputs: the desired pitch angle θ_d, the desired roll angle ϕ_d, the desired vertical velocity $\dot{z}_d$ and the desired yaw angular velocity r_d, and then it calculates the required four motor forces. In this paper, we assume that all the states of the quadrotor are measured. We first use standard, nonlinear controllers to stabilize the z-value of the vehicle to a fixed value $z = z_d$, and the yaw angle to zero ($\psi_d = 0$). Then, we manipulate θ_d and ϕ_d to control the vehicle's motion in the x-, y-directions. Assuming that the nonlinear controller successfully stabilizes the vehicle at $z = z_d$ and $\psi = \psi_d = 0$, we can assume $\ddot{z} = 0$ and $\psi = 0$ in the translational dynamics, and then the dynamics can be reduced to $\ddot{x} = g\, tan(\theta)$, $\ddot{y} = -g\, tan(\phi)/cos(\theta)$. Now we linearize the reduced dynamics, so that we can apply the proposed algorithm to calculate safe speed profiles in the x-, y-directions. To that end, let $v_1 := g\, tan(\theta_d)$ and $v_2 := -g\, tan(\phi_d)/cos(\theta_d)$. Equivalently, $\theta_d = arctan(v_1/g)$ and $\phi_d = arctan(-v_2 cos(\theta_d)/g)$. If the onboard controller successfully stabilizes ϕ and θ to ϕ_d and θ_d, respectively, then the dynamics become

$$\ddot{x} = v_1,\ \ddot{y} = v_2, \tag{4}$$

which are decoupled double integrators. Since the onboard controller typically operates much faster than the position controllers (200 Hz and 70 Hz, respectively, in our experiments), it is reasonable to assume that ϕ and θ are stabilized to ϕ_d and θ_d quickly, and (4) holds approximately. We have the following constraints on the inputs to the onboard controller: $|\phi_d| \leq 0.32$ rad, and $|\theta_d| \leq 0.32$ rad. It can be verified that if $|v_i| \leq 3.247$, $i \in \{1, 2\}$, then the constraints on ϕ_d and θ_d are satisfied.

Based on the above, our role reduces to constructing for (4) IBC regions under the limits $|v_i| \leq 3.247$, $i \in \{1, 2\}$. Suppose, for instance, that the position safety constraints are: $-2 \leq x \leq 2$ and $-2 \leq y \leq 2$. Similar to Example 5.1, we use Algorithm 1 to construct the IBC polytopes. Following Remark 5.2, one can always scale the obtained IBC polytopes, so that the IBC property holds on the scaled polytopes using inputs within the actuation limits. Indeed, for the double integrator system, it can be shown that one can only scale the velocity components of the states, and end up with new polytopes satisfying the IBC property under the actuation limits. For instance, Fig. 2 shows the IBC region for the dynamics in the x-direction under $|v_1| \leq 3.247$. If one limits the speed at x, $-2 \leq x \leq 2$, to the safe speed range, then there exist feasible inputs that keep the state trajectory inside the IBC region. Moreover, we provide in the proof of Theorem 5.2 a constructive method for synthesizing a PWL feedback that keeps the state trajectories inside the IBC regions. Indeed, one can further simplify the computation of the PWL control law in this application. By selecting P in Step 1 of Algorithm 1 to be a symmetric square around the origin, the constructed X under $v_1 \in [-v_{max}, v_{max}]$ will have the shape in Fig. 1, and it is symmetric around the origin. Let $x_1 = x$, $x_2 = \dot{x}$, and $x_{2,max}$ denote the maximum x_2-component in X. For the vertices on $\mathcal{O}$, the ones with zero-velocity components,

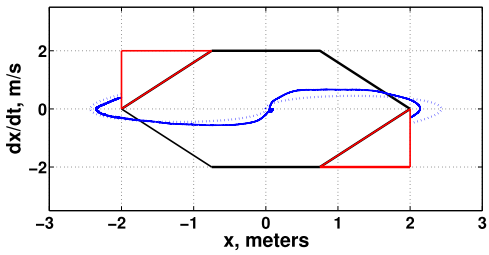

Fig. 3. The state trajectories connecting the origin to some points outside the IBC region (dotted lines: simulations; solid lines: experiments). Points within the red triangles are not reachable from other points in the safe region within the region itself.

we select $v_1 = 0$. Then, we select $v_1 = -v_{max}$ and $v_1 = v_{max}$ for the vertices with positive and negative velocity components x_2, respectively, which satisfy the invariance conditions of X. Using this selection of the inputs at the vertices and the triangulation in Fig. 5, it can be verified that the PWL feedback reduces to a linear feedback $v_1 = -\frac{v_{max}}{x_{2,max}} x_2$, which can be simply calculated online.

Fig. 2 shows samples of the state trajectories, under the proposed feedback in the previous paragraph, initiated at different critical states inside the IBC region (blue trajectories). For all the shown initial conditions in the IBC region, the proposed feedback successfully keeps the state trajectories in the IBC region, and prevents the violation of the constraints. After decelerating the vehicle to zero velocity, one can apply a robust hovering controller to keep the vehicle in place. Fig. 2 also shows two cases where the vehicle is initiated at high initial velocities, outside the safe speed profile, in the direction of $x = 2$ (red trajectories). For these cases, the proposed feedback, built based on the vehicle's actuation limits, cannot decelerate the vehicle before collision.

In the second experiment, we compare the proposed safe speed profile to the ones that can be obtained by intuition or by the controlled invariance property. One can argue that the states in the red triangles in Fig. 3 should be included in the safe position–speed region since starting from any state in the red triangles, the position constraints are not violated. However, these states in the red triangles are not reachable from all other states inside the safe region within the region itself. Hence, our algorithm automatically truncates these red triangles to ensure full controllability on the safe region. In Fig. 3, we show the state trajectories of connecting the origin to some points in the red triangles. The dotted blue trajectories are obtained from simulation by applying the standard, open-loop control law of connecting two states based on the control Gramian (equation (15) of Helwa and Caines (2014a), with $t_f = 10$ s). The solid blue trajectories are obtained experimentally by applying similar acceleration profiles to the real system. One can see that the vehicle cannot reach the points in the red triangles without violating the safety position constraints. Hence, the points of reference trajectories should be always selected inside the IBC region to ensure that they can be reached from other safe states within the safe region.

One advantage of our proposed algorithm is that it is computationally efficient, and this enables us to implement the algorithm in real time at 70 Hz to achieve dynamic obstacle avoidance when dynamic obstacles intersect with the vehicle's path. In particular, the position constraints are updated based on the detected obstacles, and then we calculate corresponding safe speed profiles. On the constructed IBC region, we then utilize the calculated linear feedback to keep the vehicle's trajectories within the IBC region, as discussed before. In the third experiment, we let the vehicle track a sinusoidal reference in the y-direction with a frequency of 0.1 Hz, while stabilizing the x-value to $x_d = 0$ and keeping a constant height. We then run another vehicle, our dynamic obstacle, to track a sinusoidal reference in the x-direction with a frequency of 0.1 Hz, while stabilizing the y-value to $y_d = 0$ and again keeping a constant height. The two vehicles collide if their x-, y-coordinates coincide. Fig. 4(a) shows that the proposed algorithm, running on the first vehicle, prevents the collision between the two vehicles. We also illustrate in green the case where we do not run our proposed algorithm in real time. One can see that the Euclidean distance between the two vehicles drops below 0.5 m in this case, which we consider a crash given the vehicle body radius of 0.32 m. In Fig. 4(b), we repeat the same experiment after replacing the second vehicle with a random human motion. Our proposed algorithm again prevents the collision. A demo is at: http://tiny.cc/ibc-quad.

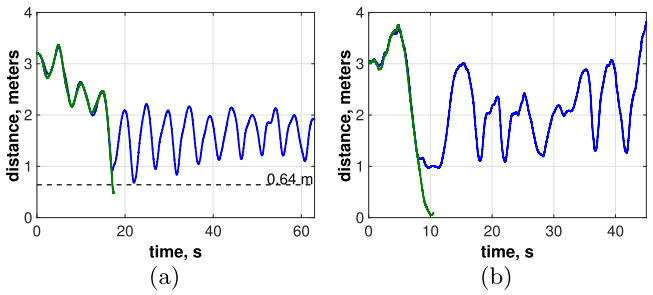

Fig. 4. The Euclidean distance between the x-, y-coordinates of the vehicle center and the dynamic obstacle ($\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$). Blue line: with the proposed algorithm for updating the safe speed profile online. Green line: without updating the safe speed profile online (the experiment was stopped after collision). (a) two vehicles; (b) moving human. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

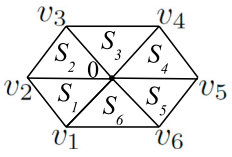

Fig. 5. An illustrative figure for the triangulation used in the proof of Theorem 5.2.

7. Conclusions

We studied the problem of constructing IBC regions for affine systems. That is, we construct safe regions in the state space within which we can fully control the given affine system using uniformly bounded inputs. After formulating the problem, we discussed the faced difficulties if one tries to directly exploit the existing results for checking IBC on given polytopes. Instead, we provided a novel, computationally efficient algorithm for constructing IBC regions, and proved its soundness. As a case study, we showed how our proposed algorithm can be useful for achieving real-time collision avoidance and for checking feasibility of reference trajectories for UAVs.

Appendix

Continuation of the Proof of Theorem 5.2. We construct a continuous PWL feedback $u_p(x)$ under which all the trajectories initiated in X° tend to $\mathcal{O}$ through X°. At a vertex $\bar{v} \in \mathcal{O}$, select input $\bar{u}$ such that $A\bar{v} + B\bar{u} = 0$, which is always possible by the definition of $\mathcal{O}$. Next, for the vertices v_i satisfying $\mathcal{B} \cap C^\circ(v_i) \neq \emptyset$, identify $\bar{b}_i \in \mathcal{B} \cap C^\circ(v_i)$. Since $\bar{b}_i \in C^\circ(v_i)$, then by definition $h_j \cdot \bar{b}_i < 0$, for all $j \in J(v_i)$. Also, since $\bar{b}_i \in \mathcal{B}$, there exists $\bar{u}_i \in \mathbb{R}^m$ such that $B\bar{u}_i = \bar{b}_i$. Now for $u_i = c_i\bar{u}_i \in \mathbb{R}^m$, where $c_i > 0$, we have $h_j \cdot (Av_i + Bu_i) = h_j \cdot Av_i + c_i h_j \cdot \bar{b}_i$, for all $j \in J(v_i)$. The second term of the right-hand side is always negative, and we can always

select $c_i > 0$ sufficiently large such that $h_j \cdot (Av_i + Bu_i) < 0$, for all $j \in J(v_i)$. The above control assignment at the vertices of X satisfies the invariance conditions, and for the vertices having $v_i \notin \mathcal{O}$ and $\mathcal{B} \cap C^\circ(v_i) \neq \emptyset$, it satisfies the invariance conditions strictly (with strict inequalities). At $x = 0$, select $u = 0$. We construct a special triangulation of X using the point set $\{v_1, \ldots, v_p, 0\}$, where $\{v_1, \ldots, v_p\}$ are the vertices of X, such that if S_i is an n-dimensional simplex in the triangulation, then $0 \in S_i$ is a vertex of S_i. This can be carried out by triangulating each facet of X, F_j, into $(n-1)$-dimensional simplices, and then taking the convex hull of $0 \in X^\circ$ and the $(n-1)$-dimensional simplices to form a triangulation of X consisting of n-dimensional simplices S_i with the desired property. Fig. 5 shows a 2D illustration of the triangulation. Based on the control values selected at $\{v_1, \ldots, v_p, 0\}$, one can always construct on each simplex S_i a unique affine feedback $k_i x + g_i$. Moreover, $[k_i \ g_i]^T = [\bar{V} \ \bar{1}]^{-1} \bar{w}$, where $\bar{V}$ is a matrix whose rows are the transpose of the vertices of $\mathcal{S}_i$, $\bar{1}$ is a column of ones, and $\bar{w}$ is a column of the transpose of the selected inputs at the vertices of $\mathcal{S}_i$ (Habets & van Schuppen, 2004). Since $u = 0$ at $x = 0$ by assignment and $0 \in \mathcal{S}_i$, then $g_i = 0$; that is, the feedback on each $\mathcal{S}_i$ is linear. It can be easily shown that the overall control law is a continuous PWL feedback, denoted by $u_p(x)$, and by a simple convexity argument, $u_p(x)$ satisfies the invariance conditions of X at every $x \in \partial X$.

Let $f(x) := Ax + Bu_p(x)$. Since 0 is a vertex in each S_i, $f(0) = 0$ and $f(x)$ is linear on each S_i, then the vector field on $\partial(\lambda X)$ represents λ-scaled vectors of the vector field on ∂X for any $\lambda \in (0, 1)$. Therefore, $u_p(x)$ satisfies the invariance conditions of λX for any $\lambda \in (0, 1)$, and so for any $x_0 \in \lambda X$, $\phi(x_0, t, u_p) \in \lambda X$ for all $t \geq 0$.

We next show that for every $x_0 \in X^\circ$, $\phi(x_0, t, u_p) \to \mathcal{O}$ as $t \to \infty$, which implies by a simple argument that we can steer the trajectories to an ϵ-neighborhood of $\mathcal{O}$ in finite time, where $\epsilon > 0$ can be selected arbitrarily small. Since $0 \in X^\circ$ by assumption, it is known that X can be expressed as $X = \{x \in \mathbb{R}^n : n_i \cdot x \leq 1, \ i = 1, \ldots, r\}$, where $n_i \in \mathbb{R}^n$, $n_i \cdot x = 1$ if $x \in F_i$ and $n_i \cdot x < 1$ if $x \in X$, $x \notin F_i$. We define $V(x) = \max_{i \in \{1, \ldots, r\}} n_i \cdot x$. Notice that if $x \in \partial X$, then $V(x) = 1$. Similarly, if $x \in \partial(\lambda X)$ for $\lambda \in (0, 1)$, then $V(x) = \lambda$. One can show that $V(x)$ is locally Lipschitz, and its upper right Dini derivative $D_f^+ V(x) = \max_{i \in I(x)} n_i \cdot (Ax + Bu_p(x))$, where $I(x) = \{i \in \{1, \ldots, r\} : n_i \cdot x = V(x)\}$ (Danskin, 1966). With the aid of invariance conditions, it is shown in Lemma 5.3 of Helwa and Caines (2014a), which also applies to non-simplicial polytopes, that $D_f^+ V(x) \leq 0$ for each $x \in X$. We hereby show that additionally $\{x \in X : D_f^+ V(x) = 0\} \subset \mathcal{O}$. Notice that for a vertex $v_i \in \mathcal{O}$, $f(v_i) = 0$ by assignment, and so $D_f^+ V(v_i) = 0$. Next, the rest of the vertices of X satisfy $\mathcal{B} \cap C^\circ(v_i) \neq \emptyset$ by assumption, and we assigned the control inputs at these vertices to satisfy the invariance conditions strictly. Thus, $n_j \cdot (Av_i + Bu_p(v_i)) < 0$, for all $j \in J(v_i)$. Note that $j \in I(v_i)$ if by definition $n_j \cdot v_i = V(v_i) = 1$, i.e., $v_i \in F_j$. Then by the strict invariance conditions, we have $n_j \cdot (Av_i + Bu_p(v_i)) < 0$ for all $j \in I(v_i)$, and so $D_f^+ V(v_i) < 0$ for all the vertices $v_i \notin \mathcal{O}$. Let $\bar{x} \in \partial X$ be arbitrary, and suppose that $\bar{x} \in S_k$. Let $S_{\bar{x}}$ denote the smallest sub-simplex of S_k such that $\bar{x} \in S_{\bar{x}}^\circ$, the relative interior of $S_{\bar{x}}$. Since $\bar{x} \in S_{\bar{x}}^\circ$, we can write $\bar{x} = \sum_s \alpha_s v_s$, where $\alpha_s > 0$, $\sum_s \alpha_s = 1$, and v_s are the vertices of $S_{\bar{x}}$, which are a subset of the vertices of the n-dimensional simplex S_k. Since the vector field $f(x)$ is linear on the simplex S_k by construction, we have $f(\bar{x}) = \sum_s \alpha_s f(v_s)$. We now study $D_f^+ V(\bar{x})$. It is straightforward to show $I(\bar{x}) \subset I(v_s)$ for every vertex $v_s \in S_{\bar{x}}$. Then, $D_f^+ V(\bar{x}) = \max_{i \in I(\bar{x})} n_i \cdot \sum_s \alpha_s f(v_s) \leq \sum_s \alpha_s \max_{i \in I(\bar{x})} n_i \cdot f(v_s) \leq \sum_s \alpha_s \max_{i \in I(v_s)} n_i \cdot f(v_s) = \sum_s \alpha_s D_f^+ V(v_s)$. Since $\alpha_s > 0$ and $D_f^+ V(v_s) \leq 0$ for every s, then $D_f^+ V(\bar{x}) = 0$ only if $D_f^+ V(v_s) = 0$ for all the vertices $v_s \in S_{\bar{x}}$, which happens only if $v_s \in \mathcal{O}$ for every vertex $v_s \in S_{\bar{x}}$. For this case, since the set $\mathcal{O}$ is affine, then $\bar{x} \in \mathcal{O}$. To sum up, for any $x \in \partial X$, if $D_f^+ V(x) = 0$, then $x \in \mathcal{O}$. Since the vector field on $\partial(\lambda X)$ represents λ-scaled vectors of the vector field on ∂X for all $\lambda \in (0, 1)$, it can be easily shown that for any $x \in X$, if $D_f^+ V(x) = 0$, then $x \in \mathcal{O}$, i.e., $\{x \in X : D_f^+ V(x) = 0\} \subset \mathcal{O}$. Recall that $D_f^+ V(x) \leq 0$ for all $x \in X$. By LaSalle's Invariance Principle, we know that the trajectories $\phi(x_0, t, u_p)$ tend to $\{x \in X : D_f^+ V(x) = 0\} \subset \mathcal{O}$ as $t \to \infty$. Combining this with the proof in the previous paragraph, for any $x_0 \in X^\circ$, $\phi(x_0, t, u_p)$ eventually tends to $\mathcal{O}$ through X°.

References

Aswani, A., Gonzalez, H., Sastry, S. S., & Tomlin, C. (2013). Provably safe and robust learning-based model predictive control. *Automatica*, *49*(5), 1216–1226.

Athanasopoulos, N., Bitsoris, G., & Lazar, M. (2014). Construction of invariant polytopic sets with specified complexity. *International Journal of Control*, *87*(8), 1681–1693.

Blanchini, F. (1999). Set invariance in control. *Automatica*, *35*(11), 1747–1767.

Brammer, R. F. (1972). Controllability in linear autonomous systems with positive controllers. *SIAM Journal on Control*, *10*(2), 339–353.

Broucke, M. E. (2010). Reach control on simplices by continuous state feedback. *SIAM Journal on Control and Optimization*, *48*(5), 3482–3500.

Caines, P. E., & Wei, Y. J. (1995). The hierarchical lattices of a finite machine. *Systems and Control Letters*, *25*(4), 257–263.

Caines, P. E., & Wei, Y. J. (1998). Hierarchical hybrid control systems: a lattice theoretic formulation. *IEEE Transactions on Automatic Control*, *43*(4), 501–508.

Danskin, J. M. (1966). The theory of max–min, with applications. *SIAM Journal on Applied Mathematics*, *14*(4), 641–664.

Dorea, C. E. T., & Hennet, J.-C. (1999). (A, B)-invariance conditions of polyhedral domains for continuous-time systems. *European Journal of Control*, *5*, 70–81.

Filippov, A. F. (1988). *Differential equations with discontinuous right hand sides*. Kluwer Academic Publishers.

Frew, E., & Sengupta, R. (2004). Obstacle avoidance with sensor uncertainty for small unmanned aircraft. In *IEEE conf. on decision and control* (pp. 614-619), Bahamas.

Grzonka, S., Grisetti, G., & Burgard, W. (2012). A fully autonomous indoor quadrotor. *IEEE Transactions on Robotics*, *28*(1), 90–100.

Habets, L. C. G. J. M., & van Schuppen, J. H. (2004). A control problem for affine dynamical systems on a full-dimensional polytope. *Automatica*, *40*(1), 21–35.

Heemels, W. P. M. H., & Camlibel, M. K. (2007). Controllability of linear systems with input and state constraints. In *IEEE conf. on decision and control* (pp. 536-541), New Orleans.

Helwa, M. K. (2015). In-block controllability of controlled switched linear systems on polytopes. *IFAC-Papers OnLine*, *48*(27), 7–12 Elsevier.

Helwa, M. K., & Broucke, M. E. (2013). Monotonic reach control on polytopes. *IEEE Transactions on Automatic Control*, *58*(10), 2704–2709.

Helwa, M. K., & Broucke, M. E. (2015). Flow functions, control flow functions, and the reach control problem. *Automatica*, *55*, 108–115.

Helwa, M. K., & Caines, P. E. (2014a). In-block controllability of affine systems on polytopes. In *IEEE conf. on decision and control* (pp. 3936-3942), Los Angeles.

Helwa, M. K., & Caines, P. E. (2014b). Relaxed in-block controllability of affine systems on polytopes. In *IEEE conf. on decision and control* (pp. 3943-3949), Los Angeles.

Helwa, M. K., & Caines, P. E. (2014c). Hierarchical control of piecewise affine hybrid systems. In *IEEE conf. on decision and control* (pp. 3950-3956), Los Angeles.

Helwa, M. K., & Caines, P. E. (2015a). Epsilon controllability of nonlinear systems on polytopes. In *IEEE conf. on decision and control* (pp. 252-257), Osaka.

Helwa, M. K., & Caines, P. E. (2015b). On the construction of in-block controllable covers of nonlinear systems on polytopes. In *IEEE conf. on decision and control* (pp. 276-281), Osaka.

Helwa, M. K., & Caines, P. E. (2017). In-block controllability of affine systems on polytopes. *IEEE Transactions on Automatic Control*, *62*(6), 2950–2957.

Helwa, M. K., & Schoellig, A. P. (2016). On the construction of safe controllable regions for affine systems with applications to robotics. In *IEEE conf. on decision and control* (pp. 3000-3005), Las Vegas.

Hubbard, P., & Caines, P. E. (2002). Dynamical consistency in hierarchical supervisory control. *IEEE Transactions on Automatic Control*, *47*(1), 37–52.

Kant, K., & Zucker, S. W. (1986). Toward efficient trajectory planning: the path-velocity decomposition. *The International Journal of Robotics Research*, *5*(3), 72–89.

Ostafew, C. J., Schoellig, A. P., Barfoot, T. D., & Collier, J. (2014). Speed daemon: experience-based mobile robot speed scheduling. In *The Canadian conf. on computer and robot vision* (pp. 56-62), Montreal.

Palafox, O. M., & Spong, M. W. (2009). Bilateral teleoperation of a formation of nonholonomic mobile robots under constant time delay. In *IEEE/RSJ intl. conf. on intelligent robots and systems* (pp. 2821-2826), St. Louis.

Qin, S. J., & Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, *11*(7), 733–764.

Rockafellar, R. T. (1970). *Convex analysis*. Princeton, New Jersey: Princeton University Press.

Rodriguez-Seda, E. J., Stipanovic, D. M., & Spong, M. W. (2011). Lyapunov-based cooperative avoidance control for multiple Lagrangian systems with bounded sensing uncertainties. In *IEEE conf. on decision and control* (pp. 4207-4213), Orlando.

Rodriguez-Seda, E. J., Tang, C., Spong, M. W., & Stipanovic, D. M. (2014). Trajectory tracking with collision avoidance for nonholonomic vehicles with acceleration constraints and limited sensing. *The International Journal of Robotics Research*, *33*(12), 1569–1592.

Schoellig, A. P., Hehn, M., Lupashin, S., & D'Andrea, R. (2011). Feasibility of motion primitives for choreographed quadrocopter flight. In *The American control conf.* (pp. 3843-3849), San Francisco.

Sontag, E. D. (1984). An algebraic approach to bounded controllability of linear systems. *International Journal of Control*, *39*(1), 181–188.

Thuan, L. Q., & Camlibel, M. K. (2014). Controllability and stabilizability of a class of continuous piecewise affine dynamical systems. *SIAM Journal of Control and Optimization*, *52*(3), 1914–1934.

Toker, O., & Ozbay, H. (1995). On the NP-hardness of solving bilinear matrix inequalities and simultaneous stabilization with static output feedback. In *The American control conf.* (pp. 2525-2526), Seattle.

Mohamed K. Helwa received his B.Sc. and M.Sc. degrees in Electrical Engineering from Cairo University in 2006 and 2009 respectively, and then he received his Ph.D. degree in Electrical & Computer Engineering from the University of Toronto in 2013. He worked as a part-time control engineer at Invensys Process Systems from 2007 to 2009, as a postdoctoral fellow at the Centre for Intelligent Machines, McGill University from 2013 to 2015, and as a lecturer at the University of Toronto in Winter 2016. Since 2016, he has been working as a postdoctoral fellow at the Dynamic Systems Lab, Robotics Group, University of Toronto Institute for Aerospace Studies. He is also with the Electrical Power and Machines Department, Cairo University, and is affiliated with Vector Institute for Artificial Intelligence. His research interests include control design for safety-critical systems, hierarchical control strategies, safe and data-efficient learning-based control strategies, and various applications of control systems and machine learning to mechatronic systems, including robotics, electric vehicles, aerial vehicles, and autonomous cars. Further information may be found at https://sites.google.com/site/mohamedkhelwacom/.

Angela Schoellig is an Assistant Professor at the University of Toronto Institute for Aerospace Studies and an Associate Director of the Centre for Aerial Robotics Research and Education at the University of Toronto. She conducts research at the interface of robotics, controls, and machine learning. Her goal is to enhance the performance, safety, and autonomy of robots by enabling them to learn from past experiments and from each other. She is a recipient of a Sloan Research Fellowship, a Ministry of Research, Innovation & Science Early Researcher Award, and a Connaught New Researcher Award. Her team won the 2018 GM/SAE AutoDrive Challenge, a North-America-wide self-driving competition. She is one of MIT Technology Review's Innovators Under 35 (2017), one of Robohub's "25 women in robotics you need to know about (2013)," winner of MIT's 2015 Enabling Society Tech Competition, a 2015 finalist in Dubai's $1 million "Drones for Good" competition, and the youngest member of the 2014 Science Leadership Program, which promotes outstanding scientists in Canada.

Angela received her Ph.D. from ETH Zurich, Switzerland, in 2013, and holds both an M.Sc. in Engineering Science and Mechanics from the Georgia Institute of Technology (2007) and a Masters degree in Engineering Cybernetics from the University of Stuttgart, Germany (2008). Her Ph.D. was awarded the ETH Medal and the 2013 Dimitris N. Chorafas Foundation Award (one of 35 worldwide). Further information may be found at http://www.schoellig.name.