Reachability Analysis of Discrete-Time Systems With Disturbances

Saša V. Raković, Eric C. Kerrigan, *Member, IEEE*, David Q. Mayne, *Fellow, IEEE*, and John Lygeros, *Member, IEEE*

***Abstract*—This paper presents new results that allow one to compute the set of states that can be robustly steered in a finite number of steps, via state feedback control, to a given target set. The assumptions that are made in this paper are that the system is discrete-time, nonlinear and time-invariant and subject to mixed constraints on the state and input. A persistent disturbance, dependent on the current state and input, acts on the system. Existing results are not able to address state- and input-dependent disturbances and the results in this paper are, therefore, a generalization of previously published results. One of the key aims of this paper is to present results such that one can perform the relevant set computations using polyhedral algebra and computational geometry software, provided the system is piecewise affine and the constraints are polygonal. Existing methods are only applicable to piecewise affine systems that either have no control inputs or no disturbances, whereas the results in this paper remove this limitation. Some simple examples are also given that show that, even if all the relevant sets are convex and the system is linear, convexity of the set of controllable states cannot be guaranteed.**

***Index Terms*—Controllability, control systems, nonlinear systems, Piecewise affine systems, reachability analysis, robustness, set invariance.**

I. Introduction

THE problems of reachability, invariance and control invariance for discrete-time systems have been extensively studied in the literature for over four decades (see [1]–[6] for some seminal papers on the subject). Recently these problems have attracted renewed attention, partly because improvements in computational capabilities have made it possible to implement the algorithms for systems of practical interest (see, for instance, an excellent survey paper [7] for more details and a set of relevant references). Another reason for the renewed interest in these problems is the emergence of new classes of practically important systems, such as hybrid systems. These are systems whose states, inputs and outputs can take on values from both a countable set (e.g., the set of integers) as well as an uncountable set (e.g., the set of real numbers). In recent years, invariance and reachability problems for classes of hybrid systems have been studied by a number of authors [8]–[15].

One class of systems that, to the authors' knowledge, has received relatively little attention are systems with mixed constraints on the states, control inputs and disturbances. When this class of systems is treated, it is often with an insufficient amount of detail and overly conservative approximations. Systems with mixed state, control and disturbance constraints may arise in practice for a number of reasons.

1) When modeling systems with physical constraints. Here the model must reflect the fact that the constraints will be satisfied by all evolutions of the system, whatever the control inputs and disturbances.
2) When designing controllers to meet safety or performance specifications, i.e., to ensure that the state of the system remains in a certain region of the state space. Safety and performance specifications may be violated if the inputs are not chosen properly.

A couple of simple examples illustrate the point. Consider the following discrete-time model for the longitudinal motion of a car on a highway:

$$\begin{bmatrix} x^+ \\ v^+ \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w$$

where $x \in \mathrm{IR}$ represents the position of the car, $v \in \mathrm{IR}$ its velocity, $u \in [\underline{u}, \overline{u}]$ represents the control acceleration applied by the engine or brakes, and $w \in [\underline{w}, \overline{w}]$ a disturbance acceleration due to wind. It is assumed that $\underline{u} < 0 < \overline{u}$ and $\underline{w} < \overline{w}$. For simplicity, all other constants have been normalized to 1.

One would like to capture the situation where the vehicle is prevented from going backward. This is a reasonable requirement in many cases (e.g., on a highway) and is very easy to implement in practice (assuming that the wind is incapable of pushing the car backward when the brakes are applied one could simply disallow the reverse gear). This can be captured by the hard state constraint $v \geq 0$. To enforce this constraint, the model needs to incorporate the additional state-dependent constraint $v + u + w \geq 0$ on the inputs (control and disturbance).

For another example, consider the following piecewise affine system:

$$x^+ = Ax + B\mathrm{sat}_u(u + Ew) \tag{1}$$

which is subject to a bounded disturbance $w \in \mathcal{W}$. The function $\mathrm{sat}_u(\cdot)$ models physical saturation limits on the input. Assuming that these saturation limits are symmetric and have unit

Manuscript received December 23, 2004; revised August 22, 2005. Recommended by Associate Editor C. T. Abdallah. This research was supported in part by the Engineering and Physical Sciences Research Council and the Royal Academy of Engineering, U. K., and in part by the European Commission under the HYCON Network of Excellence Contract FP6-IST-511368.

S. V. Raković and D. Q. Mayne are with the Department of Electrical and Electronic Engineering, Imperial College London SW7 2BT, U.K. (e-mail: sasa.rakovic@imperial.ac.uk; d.mayne@imperial.ac.uk).

E. C. Kerrigan is with the Department of Electrical and Electronic Engineering and the Department of Aeronautics, Imperial College London SW7 2BT, U.K.(e-mail: e.kerrigan@imperial.ac.uk).

J. Lygeros is with the Systems and Control Division, Department of Electrical and Computer Engineering, University of Patras, Patras GR26500, Greece (e-mail: lygeros@ee.upatras.gr).

Digital Object Identifier 10.1109/TAC.2006.872835

magnitude, an equivalent way of modeling (1) is to treat it as linear system with an input-dependent disturbance, i.e., letting

$$x^{+} = Ax + Bu + BEw \tag{2}$$

where the control is constrained to

$$\mathcal{U} := \{u \mid \|u\|_{\infty} \leq 1\} \tag{3}$$

and the input-dependent disturbance satisfies $w \in \mathcal{W}(u)$, where

$$\mathcal{W}(u) := \{w \in \mathcal{W} \mid \|u + Ew\|_{\infty} \leq 1\}. \tag{4}$$

Another common reason why state- and input-dependent disturbances arise in practice is when it is known that the uncertainty of a model is greater in certain regions of the state-input space than in other regions. For example, when a nonlinear model is linearized, the uncertainty gets larger the further one gets from the point of linearization. This uncertainty can be modeled as a state- and input-dependent disturbance, where the size of the disturbance decreases the closer one gets to the point of linearization. A state- and input-dependent disturbance model will therefore allow one to obtain less conservative results than if one were to assume that the disturbance is independent of the state and input.

Another example when one can model uncertainty as a state- and input-dependent disturbance is when there is parametric uncertainty present in the model. For example, if there is uncertainty in the pair (A, B) in (2), then one can think of the uncertainty as an additional state- and input-dependent disturbance. The reader is referred to [16] to see how reachability computations can be carried out for this specific class of uncertainty when the system is linear. The results in this paper can, with some effort, be used to extend the results in [16] to the class of piecewise affine systems with parametric uncertainty.

More generally, consider state variables x, control variables u and disturbance variables w, taking values in the sets X, U, and W, respectively. Consider dynamic constraints on these variables of the form

$$x_{k+1} = f(x_k, u_k, w_k) \text{ and } (x_k, u_k, w_k) \in \Upsilon \tag{5}$$

where $\Upsilon \subseteq X \times U \times W$ and $f : \Upsilon \rightarrow X$. Here, Υ is assumed to capture the physical, state-dependent constraints on the control and disturbance inputs. The goal is to develop methods for designing controllers for this class of dynamical systems.

Though fairly general results exist that can be applied to a large class of nonlinear discrete-time systems, to our knowledge, none of these control and analysis algorithms are capable of explicitly dealing with this class of problems. For example, most authors assume that the disturbance is not dependent on the state and input—the only paper which addresses state-dependent disturbances directly (for linear systems) is [17].

The key tool that allows one to perform a reachability analysis (often also called a controllability analysis), is software for implementing the so-called *predecessor* operator, which allows one to compute the set of states that can be robustly steered (using an admissible control input) to a given target set in a single step. The predecessor operator is then called in a recursive fashion in order to compute the set of states that can be robustly steered to the given target set in a finite number of steps.

A direct way of approximating the computation of the predecessor set is to grid the state-input-disturbance space, effectively approximating the original system by a finite state-input-disturbance system. Clearly, this approach has computational complexity drawbacks, since the computation grows exponentially with the dimension of the state, input and disturbance spaces. Moreover, even though results exist guaranteeing asymptotic convergence to the real set as the grid gets finer, in practice it is not always clear how fine or coarse the grid needs to be in order to have sufficiently accurate results.

A more elegant approach is to use symbolic algebra software and/or quantifier elimination methods [15], [18]–[20]. The idea here is to encode the predecessor computation in an appropriate system of logic using quantifiers to capture requirements that need to hold for some control actions, all disturbance, at some or for all times, etc. Computational tools [21], [22] can then be used to eliminate the quantifiers in these formulas and derive quantifier free formulas that define the set of states where the requirements hold (e.g., the predecessor set). For many classes of systems this approach is exact and does not involve any approximation. Moreover, the quantifier elimination approach is very general. In addition to linear and piecewise linear/affine systems (on which the computational methods proposed in this paper mostly apply), quantifier elimination methods can also be applied to a considerably more general class of discrete-time systems, for example systems whose dynamics and constraints are encoded by piecewise polynomial functions. The limits of the applicability of this approach to continuous-time systems are investigated in [23], where methods for using systems amenable to the quantifier elimination approach to approximate even more general classes of systems are also discussed.

The main drawback of methods based on quantifier elimination is their complexity. It is known that general purpose quantifier elimination is worst case doubly exponential in the size of the input and output data. For the classes of problems considered here and under some conditions (e.g., absence of control and/or disturbance variables), one can exploit structure present in the formulas used to encode the predecessor computation to get better performance [24], [25]. Worst case bounds are still exponential, even though the running times observed in practice are typically much faster. [26] presents the results in this line of work that are most closely related to our study. In this reference, the special structure afforded by piecewise linear functions is exploited to derive algorithms with very reasonable running times, reasonable enough to allow their application to realistic problems in network monitoring. For other cases of the application of symbolic methods to problems in control theory (equilibrium computation, stabilization, tracking) the reader is referred to [18], [27], and [28].

It is well-known that if the system $f(\cdot)$ is linear or piecewise affine and the relevant constraints sets (e.g., Υ) are polygons, then standard software for polytope manipulation can be used for reachability analysis [7], [13], [29]. There are a number of benefits that can be obtained from using computational geometry software, rather than gridding the state–space or using quantifier elimination and computer algebra packages.

- Many algorithms for performing fundamental operations on polyhedra have a computational complexity that is a *polynomial* function of the size of the input *and* output data [30]. As mentioned above, many quantifier elimination algorithms do not have this property and their computational complexity is often doubly-exponential with respect to the size of the input and/or output data. For numerical methods based on gridding the computation is typically exponential in the dimension of the state, input and disturbance spaces for fixed accuracy.
- Software for manipulating polyhedra exploit the structure of the problem, whereas gridding and general-purpose quantifier elimination packages do not always do this. See [31] for some results that show how, by exploiting the structure when computing projections of polytopes, a geometric approach can reduce the computational requirements by a number of orders of magnitude, compared to quantifier elimination methods such as Fourier elimination.
- It is often easier to visualize, understand and implement the results and exploit any structure, whereas it is not always so clear how to proceed with an approach that is not geometric.

One of the key aims of this paper is to present results such that one can perform the relevant reachability computations using polyhedral algebra and computational geometry software, provided the system is piecewise affine and the constraints are polygonal. Existing methods for piecewise affine systems are limited to systems that either have no control input or no disturbance [13], whereas the results in this paper remove this limitation. The extension of these results is not trivial; we will show, via some examples, that even if all the relevant sets are convex and the system is linear, convexity of the set of controllable states cannot be guaranteed if there are mixed constraints on the state, input and disturbance.

This paper is organized as follows. The problem definition is given in Sections II-A and B, which relates the problem definition with some well-known results on set invariance. The main result for the computation of the predecessor set is presented in Section II-C, topological properties of the predecessor set are discussed in Section II-D and special cases are discussed in Section II-E. Section III highlights the fact that the reachability analysis can be carried out using polyhedral algebra if the system is piecewise affine and the relevant sets are polygons. To validate the results, Section IV presents a few simple numerical examples. The main contributions of this paper are summarized in Section V. Appendix I contains some results regarding continuity of set-valued maps and Appendix II gives some new results that allow one to compute the set difference of (possibly nonconvex) polygons.

Note that some of the results given in this paper, namely for the case where the disturbance is independent of the state and input, were originally reported in the thesis [32, Ch. 4] and the conference papers [33] and [34]. The conference paper [35] and the thesis [36] significantly extended these results to cover the more general case of state- and input-dependent disturbances; this paper follows a similar line of development. The results in [32]–[34] are summarized in Section II-E.3.

II. General Case

To keep the notation as simple as possible and maintain a large degree of generality, we will adopt a nonlinear approach for a large part of this paper. Definitions and results for interesting special cases, for example when the system is piecewise affine or the constraints on the disturbance are independent of the state, will be introduced where appropriate.

Given two sets $A \subset \mathbb{R}^n$ and $B \subset \mathbb{R}^n$, the *reflection* of A through the origin is $-A := \{-a \mid a \in A\}$, the *complement* of A in $\mathbb{R}^n$ is $A^c := \{a \in \mathbb{R}^n \mid a \notin A\}$, the *set difference* between A and B is $A \setminus B := \{a \in A \mid a \notin B\} = A \cap B^c$, the *Minkowski set addition* of A and B is $A \oplus B := \{a + b \mid a \in A,\ b \in B\}$ and the *Pontryagin difference* between A and B is $A \ominus B := \{a \mid a + b \in A \text{ for all } b \in B\}$. Given a set $S \subseteq X \times Y \subseteq \mathbb{R}^n \times \mathbb{R}^m$, the (orthogonal) *projection* of the set S onto X is defined as $\operatorname{Proj}_X(S) := \{x \in X \mid \exists y \in Y \text{ such that } (x, y) \in S\}$. The set of nonnegative integers is denoted by $\mathbb{N} := \{0, 1, \ldots\}$.

A. Definitions

Consider the problem of controlling a nonlinear discrete-time system in the form

$$x^+ = f(x, u, w) \tag{6}$$

where x is the current state (assumed to be measured), x^+ is the state at the next time instant, u is the current input, and w is an uncertain parameter, which shall be referred to as the "disturbance," and may change from one sample to the next.

The disturbance takes on values in a set, which is dependent on the current state and input, i.e.,

$$w \in \mathcal{W}(x, u) \subset W \tag{7}$$

where $W := \mathbb{R}^p$ denotes the disturbance space. We say that the disturbance is independent of the state and input if the set $\mathcal{W}(x_1, u_1) = \mathcal{W}(x_2, u_2)$ for all $(x_1, u_1) \neq (x_2, u_2)$ and will use the notation $\mathcal{W}(x, u) = \mathcal{W}$ to denote this fact. A disturbance that is dependent only on the state or input is defined in a similar fashion and the notation $\mathcal{W}(x, u) = \mathcal{W}(x)$ and $\mathcal{W}(x, u) = \mathcal{W}(u)$, respectively, will be used to denote this. We define the "nominal/no disturbance" case when $\mathcal{W}(x, u) = \{0\}$ for all (x, u). Note that the set $\mathcal{W}(x, u)$ does not directly depend on previous values of the disturbance. However, constraints of this type (used, for example, to encode rate constraints on the disturbance or the disturbance dynamics) can be included, in cases when it is possible to measure them, by appropriately extending the state to include past disturbance values. A similar comment extends to the input constraints.

The state and input are required to satisfy a set of mixed constraints

$$(x, u) \in \mathcal{Y} \subset X \times U \tag{8}$$

where $X := \mathrm{IR}^n$ is the state space and $U := \mathrm{IR}^m$ is the input space. These constraints typically arise due to physical limitations, desired levels of performance or safety considerations. Combining this constraint with the previous constraint on the disturbance, let

$$\Upsilon := \{(x,u,w) \mid (x,u) \in \mathcal{Y} \text{ and } w \in \mathcal{W}(x,u)\} \tag{9}$$

be the subset of the graph of $\mathcal{W}(\cdot)$ where the constraints on the state and input are also satisfied. In order to have a well-defined problem, we have the standing assumption that $\mathcal{W}(x,u) \neq \emptyset$ for all $(x,u) \in \mathcal{Y}$, hence

$$\mathcal{Y} = \mathrm{Proj}_{X\times U}(\Upsilon). \tag{10}$$

The state-dependent set of admissible inputs can now be defined as

$$\mathcal{U}(x) := \{u \mid (x,u) \in \mathcal{Y}\}. \tag{11}$$

The set of admissible states is then

$$\begin{aligned}\mathcal{X} &:= \{x \mid \exists u \text{ such that } (x,u) \in \mathcal{Y}\} = \mathrm{Proj}_X(\mathcal{Y}) \\ &= \{x \mid \mathcal{U}(x) \neq \emptyset\}.\end{aligned} \tag{12}$$

If the state and input constraints are not coupled, then we will use the notation $\mathcal{U}(x) = \mathcal{U}$ or $\mathcal{Y} = \mathcal{X} \times \mathcal{U}$ to denote this.

Remark 1: Note that for the case when a feedback control law $\kappa : X \to U$ is applied to (6), by considering $x^+ = g(x,w) := f(x,\kappa(x),w)$ with $w \in \overline{\mathcal{W}}(x)$ and $x \in \overline{\mathcal{X}}$, where $\overline{\mathcal{W}}(x) := \mathcal{W}(x,\kappa(x))$ and $\overline{\mathcal{X}} := \{x \mid (x,\kappa(x)) \in \mathcal{Y}\}$, the required reachability analysis follows the procedure outlined in Section II-E.

Often part of the control objective is to guarantee robust convergence to a given set, either in minimum time, some finite time or asymptotically. Let X_f denote this so-called *target set* (also often called *terminal constraint set*) and, without loss of generality, assume that

$$X_f \subseteq \mathcal{X}. \tag{13}$$

One of the key aims of this paper is to present results that allow for the computation of the set of initial states for which a time-varying state feedback control law exists such that the constraints on the state and input (8) are robustly satisfied (for all allowable disturbances) over a finite horizon and that the state is guaranteed to be in X_f at the end of the horizon.

Let $\pi := \{\mu_0(\cdot), \mu_1(\cdot), \ldots, \mu_{N-1}(\cdot)\}$ denote a control policy (sequence of control *laws*, i.e., $\mu_i : X \to U, i = 0,\ldots,N-1$) over a horizon of length N and let $\mathbf{w} := \{w_0, w_1, \ldots, w_{N-1}\}$ denote a sequence of disturbances. Also, let $\phi(\cdot; x, \pi, \mathbf{w})$ denote the solution of (6) when the state is x at time 0 (since the system is time-invariant, we can always take the current time to be zero), the control policy is π and the disturbance sequence is $\mathbf{w}$.

For a given current state x and policy π, let $\mathbf{W}(x,\pi)$ be the set of admissible disturbance sequences of length N, i.e.,

$$\begin{aligned}\mathbf{W}(x,\pi) := \{\mathbf{w} \mid w_i \in \mathcal{W}(\phi(i;x,\pi,\mathbf{w}), \mu_i(\phi(i;x,\pi,\mathbf{w}))),& \\ i = 0,1,\ldots,N-1\}.&\end{aligned} \tag{14}$$

Clearly, if the disturbance is independent of the state and input, then $\mathbf{W}(x,\pi) = \mathcal{W}^N = \mathcal{W} \times \cdots \times \mathcal{W}$ for all (x,π).

Next, let $\Pi_N(x)$ be the set of admissible policies of length N, i.e., those policies that satisfy, for all $\mathbf{w} \in \mathbf{W}(x,\pi)$, the state and control constraints (8) over the horizon $k = 0,\ldots,N-1$, and the terminal constraint

$$\phi(N; x, \pi, \mathbf{w}) \in X_f. \tag{15}$$

In other words, the set of admissible policies is defined as

$$\begin{aligned}\Pi_N(x) := \{\pi \mid (\phi(i;x,\pi,\mathbf{w}), \mu_i(\phi(i;x,\pi,\mathbf{w}))) \in \mathcal{Y},& \\ i = 0,1,\ldots,N-1,\ \phi(N;x,\pi,\mathbf{w}) \in X_f& \\ \forall\, \mathbf{w} \in \mathbf{W}(x,\pi)\}.&\end{aligned} \tag{16}$$

The set X_N is the set of initial states for which an admissible policy of length N exists (often also called the N*-step controllable set*) and is defined as

$$X_N := \{x \mid \Pi_N(x) \neq \emptyset\}. \tag{17}$$

B. Reachability Analysis and Invariant Sets

Before proceeding to give our main result, we first recall a few well-known results that link reachability analysis to the computation of invariant sets. Central to this discussion is the so-called *predecessor set* (or *one-step set*) of a given set.

Definition 1 (Predecessor Set): Given a set $\Omega \subseteq X$, the *predecessor set* $\mathrm{Pre}(\Omega)$ is the set of states for which there exists an admissible input such that, for all allowable disturbances, the successor state is in Ω, i.e.,

$$\begin{aligned}\mathrm{Pre}(\Omega) := \{x \mid \exists u \in \mathcal{U}(x) \text{ such that}& \\ f(x,u,w) \in \Omega \text{ for all } w \in \mathcal{W}(x,u)\}.&\end{aligned} \tag{18}$$

An equivalent formulation of (18) is

$$\begin{aligned}\mathrm{Pre}(\Omega) := \{x \mid \exists u \in \mathcal{U}(x) \text{ such that}& \\ f(x,u,\mathcal{W}(x,u)) \subseteq \Omega\}&\end{aligned} \tag{19}$$

where $f(x,u,\mathcal{W}(x,u)) := \{f(x,u,w) \mid w \in \mathcal{W}(x,u)\}$.

For any integer i, let X_i denote the i*-step predecessor set* to X_f, i.e., X_i is the set of states that can be steered, by a time-varying state feedback control law, to the target set X_f in i steps, for all allowable disturbance sequences while satisfying, at all times, the constraint $(x,u) \in \mathcal{Y}$. In other words, X_i is given by (17) with $N = i$. Following a standard procedure [4], the sequence of sets $\{X_i\}_{i\in \mathrm{IN}}$ may be calculated recursively as follows:

$$X_0 = X_f \tag{20a}$$

$$X_{i+1} = \mathrm{Pre}(X_i). \tag{20b}$$

Recall that a given set $\mathcal{S} \subseteq \mathcal{X}$ is defined to be *robust control invariant* [7] if for any $x \in \mathcal{S}$, there exists a $u \in \mathcal{U}(x)$ such that $f(x,u,w) \in \mathcal{S}$ for all $w \in \mathcal{W}(x,u)$. A robust control invariant set $\mathcal{C}_\infty \subseteq \mathcal{X}$ is called *maximal* in $\mathcal{X}$ if all other robust control invariant sets in $\mathcal{X}$ are contained in $\mathcal{C}_\infty$.

We are now in a position to state some important, well-known results that link the recursion in (20) to its use in the computation of invariant sets. Since it is beyond the scope of this paper to give a detailed literature review of this subject, we refer the reader to the surveys [7] and [29] for a detailed discussion. In this paper, we would like to highlight the following results.

Proposition 1 (Results on Set Invariance):

i) There exists a unique robust control invariant set $\mathcal{C}_\infty \subseteq \mathcal{X}$ that is maximal in $\mathcal{X}$, provided that $\mathcal{C}_\infty$ is nonempty.
ii) A given set X_i is robust control invariant if and only if $X_i \subseteq X_{i+1} = \text{Pre}(X_i)$.
iii) X_i is robust control invariant for all $i \in \text{IN}$ if and only if X_f is robust control invariant.
iv) If $X_f = \mathcal{X}$, then $X_{i+1} \subseteq X_i$ for all $i \in \text{IN}$ and the maximal robust control invariant set $\mathcal{C}_\infty$ satisfies $\mathcal{C}_\infty \subseteq \bigcap_{i\in\text{IN}} X_i$. Furthermore, $\mathcal{C}_\infty = X_i$ for a given $i \in \text{IN}$ if and only if $X_{i+1} = X_i$.

Remark 2: If the system has no input u, i.e., if $f(\cdot)$ is a function only of (x, w), then Proposition 1 still holds with the appropriate modifications to definitions, but with "robust control invariant" replaced with "robust positively invariant" [7].

Remark 3: Without any additional assumptions on the system or sets, it is possible to find examples for which $\mathcal{C}_\infty \neq \bigcap_{i\in\text{IN}} X_i$ if $X_f = \mathcal{X}$ [6].

It is clear that results that enable one to compute the predecessor set also allow one to compute each of the sets in the sequence $\{X_i\}_{i\in\text{IN}}$. Furthermore, as will be shown below in Corollary 2, one can also employ the predecessor operator via the recursion (20) to compute an arbitrarily close approximation to the maximal robust control invariant set $\mathcal{C}_\infty$, provided some additional compactness and continuity assumptions are satisfied. Finally, the computation of the predecessor set plays a crucial role in allowing one to compute optimal control laws for piecewise affine discrete-time systems with disturbances [34], [37], [38].

C. Main Result

As discussed in the introduction, the main aim of this paper is to provide results that allow one to use computational geometry packages for computing the predecessor set. Due to the fact that existing computational geometry software do not provide general tools for the direct elimination of the universal quantifier in an expression, one first has to obtain an equivalent expression for the predecessor set that only contains the existential quantifier. The elimination of the existential quantifier can then be achieved by computing the projection of an appropriately defined set. Of course, any suitable quantifier elimination software may also be used to compute the projection. However, as mentioned in the Introduction, we are not aware of quantifier elimination methods with a computational complexity bound that is a polynomial function of the input and output data, whereas computational geometry methods exist with polynomial complexity bounds.

Before proceeding to state our main result, we define

$$\Sigma := \{(x,u) \in \mathcal{Y} \mid f(x,u,w) \in \Omega \text{ for all } w \in \mathcal{W}(x,u)\} \tag{21}$$

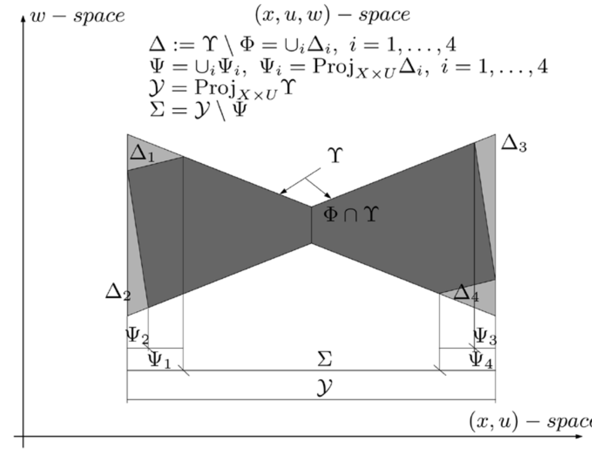

Fig. 1. Graphical illustration of Theorem 1.

the set of admissible state-input pairs for which the state of the system at the next sample instant is in a given set Ω for all admissible disturbances, and

$$\Phi := f^{-1}(\Omega) := \{(x,u,w) \mid f(x,u,w) \in \Omega\} \tag{22}$$

the set of state-input-disturbance triplets for which the state of the system evolves to a given set Ω at the next time instant.

Note that the sets Σ and Φ are also functions of the set Ω as evident from their definitions; however, in order to simplify notation in the sequel of this paper we simply write Σ and Φ but we bear in mind that $\Sigma = \Sigma(\Omega)$ and $\Phi = \Phi(\Omega)$.

We are now in a position to state our main result, originally presented in [35] and [36].

Theorem 1 (Predecessor Set): $\text{Pre}(\Omega)$, the set of states that are robustly controllable to Ω in one step, is given by

$$\text{Pre}(\Omega) = \text{Proj}_X(\Sigma) \tag{23}$$

where Σ is given by

$$\Sigma = \mathcal{Y} \setminus \text{Proj}_{X\times U}(\Upsilon \setminus \Phi). \tag{24}$$

Proof: A graphical interpretation of the proof is given in Fig. 1.

From the definition of the set difference

$$\Upsilon \setminus \Phi = \{(x,u,w) \in \Upsilon \mid f(x,u,w) \notin \Omega\} \tag{25}$$

so that

$$\begin{aligned}\text{Proj}_{X\times U}(\Upsilon \setminus \Phi) = \{(x,u) \in \text{Proj}_{X\times U}(\Upsilon) \mid \exists w \in \mathcal{W}(x,u)\\ \text{such that } f(x,u,w) \notin \Omega\}.\end{aligned} \tag{26}$$

It follows that

$$\begin{aligned}&\text{Proj}_{X\times U}(\Upsilon) \setminus \text{Proj}_{X\times U}(\Upsilon \setminus \Phi)\\ &\quad = \{(x,u) \in \text{Proj}_{X\times U}(\Upsilon) \mid f(x,u,w) \in \Omega\\ &\qquad \text{for all } w \in \mathcal{W}(x,u)\}.\end{aligned} \tag{27}$$

The proof is completed by noting that

$$\begin{aligned}\text{Proj}_X(\Sigma) &= \{x \mid \exists u \text{ such that } (x,u) \in \mathcal{Y} \text{ and}\\ &\qquad f(x,u,w) \in \Omega \text{ for all } w \in \mathcal{W}(x,u)\} && \text{(28a)}\\ &= \{x \mid \exists u \in \mathcal{U}(x) \text{ such that } f(x,u,w) \in \Omega\\ &\qquad \text{for all } w \in \mathcal{W}(x,u)\} && \text{(28b)}\\ &= \text{Pre}(\Omega). && \text{(28c)}\end{aligned}$$

■

A conceptual algorithm for computing the sets $\text{Pre}(\Omega)$ and Σ is easily constructed from the above result. The required steps are outlined by the following prototype algorithm, given the sets Ω and Υ.

1) Compute the projection $\mathcal{Y} = \text{Proj}_{X\times U}(\Upsilon)$.
2) Compute the inverse map $\Phi := f^{-1}(\Omega)$.
3) Compute the set difference $\Delta := \Upsilon \setminus \Phi$.
4) Compute the projection $\Psi := \text{Proj}_{X\times U}(\Delta)$.
5) Compute the set difference $\Sigma = \mathcal{Y} \setminus \Psi$.
6) Compute the projection $\text{Pre}(\Omega) = \text{Proj}_X(\Sigma)$.

Note also that if the recursion (20) is to be implemented, a minor modification of the previous prototype algorithm is needed; in this case there is no need to recompute $\mathcal{Y} = \text{Proj}_{X\times U}(\Upsilon)$ after initialization. We also remark that the set recursion (20) allows one to compute the sets of states that can be robustly steered in N ($N \in \text{IN}$) steps to a given target set.

In order to implement the result, we clearly need software for computing inverse images, set differences and projections. Section III will show that, provided the system $f(\cdot)$ is linear or piecewise affine and the relevant sets are polygons, then the computation of the predecessor set $\text{Pre}(\Omega)$ is easily done using standard software for polytope manipulation.

D. Topological Properties of the Predecessor Set

The following assumption will be invoked where appropriate.

- **A1)** The function $(x, u, w) \mapsto f(x, u, w)$ is continuous and the set-valued map $(x, u) \mapsto \mathcal{W}(x, u)$ is continuous and bounded on bounded sets.

We refer the reader to Appendix I for a review of some basic definitions and results on set-valued functions.

Theorem 2 (Topological Properties): Suppose **A1)** holds. If Ω and $\mathcal{Y}$ are closed, then Σ is closed. If, in addition, $\mathcal{Y}$ is compact, then $\text{Pre}(\Omega)$ and Σ are also compact.

Proof: Let the set-valued map $F : \text{IR}^r \to 2^{\text{IR}^n}$ be defined as follows:

$$F(z) := \{f(z, w) \mid w \in \mathcal{W}(z)\}, \qquad z := (x, u). \tag{29}$$

By Proposition 2 in Appendix I, the set-valued function F is continuous. The set Σ, defined in (21), is given by

$$\Sigma = \{z \in \mathcal{Y} \mid F(z) \subseteq \Omega\}. \tag{30}$$

Since F is continuous and Ω is closed, it follows from Proposition 3 in Appendix I that Σ is closed (compact if, in addition, $\mathcal{Y}$ is compact). Since $\text{Pre}(\Omega) = \text{Proj}_X(\Sigma)$, it follows that $\text{Pre}(\Omega)$ is compact if Σ is compact. ∎

Corollary 1: Suppose **A1)** holds. If $\mathcal{Y}$ and the target set X_f are compact, then each set X_i, $i \in \text{IN}$, computed as in (20), is compact.

It is very useful to note that if $\mathcal{Y}$ is compact, then the previous result can be used to establish conditions under which the maximal robust control invariant set $\mathcal{C}_\infty$ is the limit of the sequence of sets $\{X_i\}_{i\in\text{IN}}$ if $X_f = \mathcal{X}$. The next result, which follows from Corollary 1 and [6, Prop. 4], makes this claim more precise.

Corollary 2: Suppose **A1)** holds. If $\mathcal{Y}$ is compact, the maximal robust control invariant set $\mathcal{C}_\infty$ is nonempty and $X_f = \mathcal{X}$, then $\mathcal{C}_\infty$ is compact and $\mathcal{C}_\infty = \bigcap_{i\in\text{IN}} X_i$. Furthermore, given any open set $\mathcal{S} \subset X$ such that $\mathcal{C}_\infty \subset \mathcal{S}$, there exists an $i^* \in \text{IN}$ such that $\mathcal{C}_\infty \subseteq X_i \subset \mathcal{S}$ for all $i \geq i^*$.

Remark 4: Note that if $X_f \neq \mathcal{X}$, then the requirement that **A1)** hold and Υ be compact is not sufficient for $X_\infty := \lim_{i\to\infty} X_i$ (where the limit is appropriately defined and assumed to exist) to be closed or compact. It is not difficult to find examples where **A1)** holds, Υ is compact and $X_f \neq \mathcal{X}$, but X_∞ is open. Clearly, it is also not difficult to find examples for which X_∞ is not equal to the maximal robust control invariant set $\mathcal{C}_\infty$.

We also remark that if Υ is not compact then the previous observations (Corollaries 1 and 2) apply directly to any (arbitrarily large) compact subset of Υ with obvious modifications.

E. Special Cases

1) Disturbance is Dependent Only on the State or Input: Consider first the simpler case when the disturbance constraint set is a function of x only, i.e., the disturbance w satisfies $w \in \mathcal{W}(x)$. The definitions of Υ and Σ in (9) and (21), respectively, and $\text{Pre}(\Omega)$ become

$$\Upsilon := \{(x, u, w) \mid (x, u) \in \mathcal{Y} \text{ and } w \in \mathcal{W}(x)\}, \tag{31}$$

$$\Sigma := \{(x, u) \in \mathcal{Y} \mid f(x, u, w) \in \Omega \text{ for all } w \in \mathcal{W}(x)\} \tag{32}$$

and

$$\text{Pre}(\Omega) := \{x \mid \exists u \in \mathcal{U}(x) \text{ such that } f(x, u, w) \in \Omega \text{ for all } w \in \mathcal{W}(x)\}. \tag{33}$$

Theorems 1 and 2 and Corollaries 1 and 2 remain true with these changes. A similar modification is needed if the disturbance constraint set is a function of u only, i.e., the disturbance w satisfies $w \in \mathcal{W}(u)$.

2) System Does Not Have an Input: Next, consider the case when $f(\cdot)$ is a function of (x, w) only, i.e., the system has no input u and $x^+ = f(x, w)$. In this case, the constraint $(x, u) \in \mathcal{Y}$ is replaced by $x \in \mathcal{X} \subset X$ and the definitions of Σ, Υ and Φ in Theorem 1, and $\text{Pre}(\Omega)$ are replaced by

$$\Upsilon := \{(x, w) \mid x \in \mathcal{X} \text{ and } w \in \mathcal{W}(x)\}, \tag{34}$$

$$\Sigma := \{x \in \mathcal{X} \mid f(x, w) \in \Omega \text{ for all } w \in \mathcal{W}(x)\}, \tag{35}$$

$$\Phi := f^{-1}(\Omega) := \{(x, w) \mid f(x, w) \in \Omega\} \tag{36}$$

and

$$\text{Pre}(\Omega) := \{x \in \mathcal{X} \mid f(x, w) \in \Omega \text{ for all } w \in \mathcal{W}(x)\}. \tag{37}$$

In other words, $\text{Pre}(\Omega)$ is now the set of admissible states such that the successor state lies in Ω for all $w \in \mathcal{W}(x)$. In this case, the conclusion of Theorem 1 becomes

$$\text{Pre}(\Omega) = \Sigma = \mathcal{X} \setminus \text{Proj}_X(\Upsilon \setminus \Phi). \tag{38}$$

As can be seen, this special case results in less computational effort, since operations are performed in lower dimensional spaces and only one projection operation is needed.

Also, in this case where there is no control input, **A1)** is replaced by the following.

- **A1')** The function $(x, w) \mapsto f(x, w)$ is continuous and the set-valued map $x \mapsto \mathcal{W}(x)$ is continuous and bounded on bounded sets.

Theorem 2 and Corollaries 1 and 2 remain true subject to the above modifications, but with "maximal robust control invariant set" in Corollary 2 replaced with "maximal robust positively invariant set."

3) Additive, Independent Disturbances: The case when the disturbance is additive and independent of the state and input deserves a detailed discussion. In this case, Theorem 1 still provides a method for computing the predecessor set. However, an alternative to Theorem 1 was originally presented in [32]–[34]. We recall the result and its proof.

Theorem 3 (Additive, Independent Disturbance): Let the disturbance be additive and independent of the state and input, i.e., $\mathcal{W}(x,u) = \mathcal{W}$ and

$$x^+ = f(x,u) + w. \tag{39}$$

The predecessor set is then given by

$$\begin{aligned}\mathrm{Pre}(\Omega) &= \mathrm{Proj}_X \left\{(x,u) \in \mathcal{Y} \mid f(x,u) \in \Omega \ominus \mathcal{W}\right\} &\text{(40a)}\\ &= \mathrm{Proj}_X \left[\mathcal{Y} \cap f^{-1}\left(\Omega \ominus \mathcal{W}\right)\right], &\text{(40b)}\end{aligned}$$

where the Pontryagin difference $\Omega \ominus \mathcal{W}$ is given by

$$\begin{aligned}\Omega \ominus \mathcal{W} &= [\Omega^c \oplus (-\mathcal{W})]^c &\text{(41a)}\\ &= X \setminus [(X \setminus \Omega) \oplus (-\mathcal{W})]. &\text{(41b)}\end{aligned}$$

Proof: It follows easily from the definitions that

$$\begin{aligned}\mathrm{Pre}(\Omega) &= \{x \mid \exists u \text{ such that } (x,u) \in \mathcal{Y}\\ &\qquad \text{and } f(x,u) + w \in \Omega \text{ for all } w \in \mathcal{W}\} &\text{(42a)}\\ &= \mathrm{Proj}_X \{(x,u) \in \mathcal{Y} \mid f(x,u) + w \in \Omega\\ &\qquad \text{for all } w \in \mathcal{W}\} &\text{(42b)}\end{aligned}$$

hence (40) is verified directly from the definition of the Pontryagin difference. Recall that the Pontryagin difference is defined as $\Omega \ominus \mathcal{W} := \{x \in \Omega \mid x + w \in \Omega \ \text{ for all } w \in \mathcal{W}\}$, hence the truth of (41) follows from:

$$\begin{aligned}\Omega \ominus \mathcal{W} &= \{x \mid \nexists w \in \mathcal{W},\ x + w \in \Omega^c\} &\text{(43a)}\\ \Leftrightarrow (\Omega \ominus \mathcal{W})^c &= \{x \mid \exists w \in \mathcal{W},\ x + w \in \Omega^c\} &\text{(43b)}\\ &= \{x \mid \exists c \in \Omega^c, w \in \mathcal{W},\ x + w = c\} &\text{(43c)}\\ &= \{c \mid \exists x \in \Omega^c, w \in \mathcal{W},\ c + w = x\} &\text{(43d)}\\ &= \{c \mid \exists x \in \Omega^c, w \in (-\mathcal{W}),\ x + w = c\} &\text{(43e)}\\ &= \Omega^c \oplus (-\mathcal{W}). &\text{(43f)}\end{aligned}$$

■

Remark 5: It is important to note that the majority of well-known results in the control literature on computing the Pontryagin difference $\Omega \ominus \mathcal{W}$, such as those in [4], [5], and [39], only consider the case when Ω is a convex polyhedron. The above result allows for the computation of the Pontryagin difference of nonconvex polygons.

Remark 6: It is interesting to note that though (41) does not appear to have been reported in the control literature, it is a well-known identity in the field of mathematical morphology [40], [41], where the Pontryagin difference $\Omega \ominus \mathcal{W}$ is often called the *erosion* of Ω by $\mathcal{W}$.

A prototype algorithm for computing the predecessor set is easily derived from Theorem 3.

1) Compute the reflection $-\mathcal{W}$.
2) Compute the complement $\Omega^c = X \setminus \Omega$ as a set difference.
3) Compute the Minkowski sum $\Gamma := \Omega^c \oplus (-\mathcal{W})$.
4) Compute the Pontryagin difference $\Omega \ominus \mathcal{W} = X \setminus \Gamma$ as a set difference.
5) Compute $\Sigma = \{(x,u) \in \mathcal{Y} \mid f(x,u) \in \Omega \ominus \mathcal{W}\}$.
6) Compute the projection $\mathrm{Pre}(\Omega) = \mathrm{Proj}_X(\Sigma)$.

Clearly, appropriate software is needed for computing the reflection, Minkowski sum, set difference, inverse image, intersection and projection of sets. This can be done for a large class of nonlinear systems by gridding the state space or by using computer algebra packages. However, as will be pointed out in Section III, one of the aims of this paper is to highlight the fact that all these operations can be done using standard polytope software, provided the system is linear or piecewise affine and the constraint sets are polygons. At this point, it is worth pointing out that it can be shown (see, for example, [36] and [38]) that

$$\Omega \ominus \mathcal{W} = [\mathrm{convh}(\Omega) \ominus \mathcal{W}] \setminus [(\mathrm{convh}(\Omega) \setminus \Omega) \oplus (-\mathcal{W})] \tag{44}$$

where $\mathrm{convh}(\Omega)$ is the convex hull of Ω; the Pontryagin difference $\mathrm{convh}(\Omega) \ominus \mathcal{W}$ is efficiently computed using the algorithm in [39] if $\mathcal{W}$ is a polytope. It is also worth pointing it out that the formula (44) is still valid if $\mathrm{convh}(\Omega)$ is replaced by any convex set $\mathcal{C}$ that contains Ω [36].

Obviously, any algorithm for computing the Pontryagin difference that is derived from (41) or (44) will result in exactly the same set. However, in practice the computational requirements depend very much on the specifics of the problem and the computational tools that are available. It may be that an algorithm derived directly from one equation is faster than an algorithm directly based on another equation. It is also not always easy to tell whether an algorithm for computing the predecessor set is more efficient if it were based on Theorem 1 or whether it were based on Theorem 3. A possible direction for further research is to find results that allow one to determine *a priori* the most efficient algorithms for computing the predecessor set, based on sensible assumptions on the data.

Finally, we conclude this section by pointing out that all of the results in this section are true if $f(\cdot)$ is a function of x only, i.e., $x^+ = f(x) + w$, provided the appropriate modifications to definitions are made. In this case, no final projection operation is necessary, since

$$\begin{aligned}\mathrm{Pre}(\Omega) &= \{x \in \mathcal{X} \mid f(x) \in \Omega \ominus \mathcal{W}\} &\text{(45a)}\\ &= \mathcal{X} \cap f^{-1}(\Omega \ominus \mathcal{W}). &\text{(45b)}\end{aligned}$$

III. Linear and Piecewise Affine Systems With Polygonal Constraint Sets

Up to now, we have deliberately not made any special assumptions on the structure of $f(\cdot)$. The main aim of this section is to point out that the computation of the predecessor set is possible using existing computational geometry software, provided $f(\cdot)$ is linear or piecewise affine and the constraint sets are polygons.

The main reason for presenting the results in Sections II-C and E in their current form, is because it is not possible to derive an algorithm for computing the predecessor set, which uses computational geometry software, directly from the definition in (18). However, Theorems 1 and 3 allow for the straightforward derivation of algorithms that can be implemented using readily available software libraries for the manipulation of polyhedra.

All the operations encountered in Sections II-C and E, such as projection, set difference, piecewise affine maps and their inverse, Minkowski sums, intersections, etc., are easily implemented using existing computational geometry software packages. The reader is referred to [31], [42]–[44] and the large literature on computational geometry for details.

Another reason for presenting the results as above, is to maintain a high degree of generality and to emphasize the structure of the results. When dealing with piecewise affine systems or nonconvex constraints, it is easy to exhaust the reader with notational details. As a consequence of the chosen style of presentation, we are now in a better position to state some basic definitions and present the main results, without having to introduce too much additional notation.

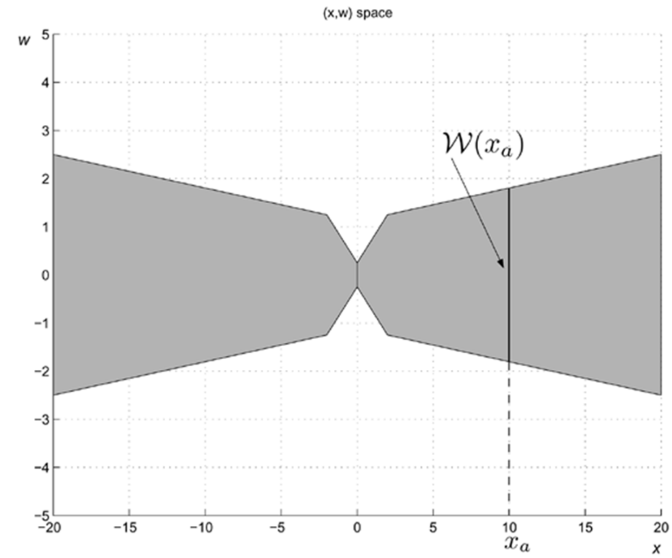

Fig. 2. Graph of $\mathcal{W}$.

A. Definitions and Notation

A *polyhedron* is the intersection of a finite number of closed and/or open halfspaces, a *polytope* is a closed and bounded (equivalently, compact) polyhedron and a *polygon* is the union of a finite number of polyhedra (and is thus not necessarily convex). A family of sets $\mathcal{P} := \{\mathcal{P}_i \mid i \in \mathcal{I}\}$ is a *(closed) polyhedral cover* of a (closed) polygon $\mathcal{X} \subseteq \mathrm{IR}^n$ if the index set $\mathcal{I}$ is finite, each $\mathcal{P}_i$ is a nonempty (closed) polyhedron and $\mathcal{X} = \cup_{i\in\mathcal{I}}\mathcal{P}_i$. Where it is useful, $\mathcal{P}^{\mathcal{X}}, \mathcal{I}^{\mathcal{X}}$ and $\mathcal{P}_i^{\mathcal{X}}$ will denote, respectively, a polyhedral cover of a polygon $\mathcal{X}$, its associated index set and the ith polyhedron in the cover.

Remark 7: It is important to discuss a few points regarding the previous definitions.

- A polyhedron is often defined in the literature to be the intersection of a finite number of *closed* halfspaces. The main reason for modifying the definition is because it allows us to considerably simplify the presentation of the results in this paper, without sacrificing rigor.
- A polyhedral cover of a polygon should not be confused with the polygon itself. The former object is a family of sets that can be used to conveniently describe the latter object, which is a single set. A given polygon may have any number of suitable polyhedral covers associated with it. This distinction between a polygon and its polyhedral cover is important when interpreting the results in this paper and implementing them with existing algorithms for polytope manipulation. For example, a *closed* polygon need not be described by a *closed* polyhedral cover; any number of members of the polyhedral cover are allowed to be neither closed nor open, provided the union of all the members is closed and equal to the polygon.
- The definition of a polyhedral cover given here is weaker than that of a so-called *polyhedral partition*, as defined in [45]. The latter object is a polyhedral cover, where the members are closed polyhedra with nonempty interiors and the interiors of the members are mutually disjoint.
- The definition of a polyhedral cover is weaker than that of a so-called *complex*, as defined in [46]. A complex is a polyhedral cover, where the members are closed polyhedra, the faces of each of the members of the cover are also members of the cover and the intersection of any two members of the cover is a face of each of them.
- Our use of the term *cover* is stronger than the one commonly used in topology, where a *cover* of a set $\mathcal{X}$ is a (possibly infinite) collection of nonempty sets $\mathcal{P} := \{\mathcal{P}_i \mid i \in \mathcal{I}\}$ such that $\mathcal{X} \subseteq \cup_{i\in\mathcal{I}}\mathcal{P}_i$. In this paper, we require equality (not the weaker condition of inclusion) and that the collection of sets is finite.

Finally, a function $f : \mathcal{X} \rightarrow \mathrm{IR}^n$ is said to be *piecewise affine* on a polyhedral cover $\mathcal{P} := \{\mathcal{P}_i \mid i \in \mathcal{I}\}$ of a polygon $\mathcal{X} \subseteq \mathrm{IR}^m$ if the restriction $f|_{\mathcal{P}_i} : \mathcal{P}_i \rightarrow \mathrm{IR}^n$ is affine for all $i \in \mathcal{I}$.

B. Main Results

In this section, we make the following assumption.

- **A2)** Υ is a polygon (hence, $\mathcal{X}$ is also a polygon) and the system $f : \Upsilon \rightarrow X$ in (6) is piecewise affine on a polyhedral cover $\mathcal{P}^{\Upsilon} := \{\mathcal{P}_i^{\Upsilon} \mid i \in \mathcal{I}^{\Upsilon}\}$ of the polygon Υ, i.e.,

$$\begin{aligned}&f(x,u,w) := A_i x + B_i u + G_i w + g_i\\&\forall\,(x,u,w) \in \mathcal{P}_i^{\Upsilon}, \qquad i \in \mathcal{I}^{\Upsilon}\end{aligned} \tag{46}$$

where for all $i \in \mathcal{I}^{\Upsilon}$, the matrices $A_i \in \mathrm{IR}^{n\times n}$, $B_i \in \mathrm{IR}^{n\times m}$, $G_i \in \mathrm{IR}^{n\times p}$, and vector $g_i \in \mathrm{IR}^n$.

Remark 8: Note that existing results on reachability analysis of discrete-time piecewise affine systems assume that either there is no control input or there is no disturbance, i.e., all the $B_i = 0$ or all the $G_i = 0$ [13]. The results in this paper allow one to remove this restriction.

For convenience, we define the functions $f_i : \mathcal{P}_i^{\Upsilon} \rightarrow X$, $i \in \mathcal{I}^{\Upsilon}$, as

$$f_i(x,u,w) := A_i x + B_i u + G_i w + g_i. \tag{47}$$

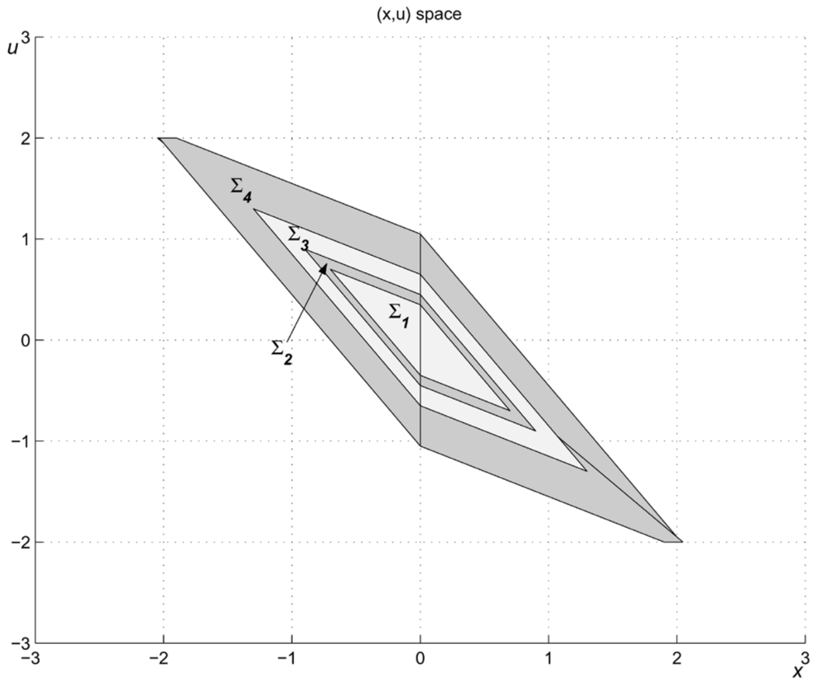

Fig. 3. Sets Σ_i for $i = 1, 2, 3, 4$.

Remark 9: Clearly, if $\mathcal{I}^\Upsilon$ has cardinality 1, then $f(\cdot)$ is affine (linear if, additionally, $g_i = 0$). Note also that, since $f(\cdot)$ is assumed to be single-valued, it follows that if $i \neq j$ and $\mathcal{P}_i^\Upsilon \cap \mathcal{P}_j^\Upsilon \neq \emptyset$, then $f_i(x,u,w) = f_j(x,u,w)$ for all $(x,u,w) \in \mathcal{P}_i^\Upsilon \cap \mathcal{P}_j^\Upsilon$.

We now give the main result of this section, where we make the assumption that the system is piecewise affine and all relevant sets are polygons. In this case, it is easy to specialize the prototype algorithms in Sections II-C and E.3 and implement them using standard computational geometry software.

Theorem 4 (Piecewise Affine Systems): Suppose **A2)** holds. If Ω is a polygon, then the predecessor set $\mathrm{Pre}(\Omega)$, as given in (18) and (23), is a polygon. Furthermore, the set Σ as given in (21) is also a polygon.

Proof: Recall the statement of Theorem 1. Let $\mathcal{P}^\Omega := \{\mathcal{P}_i^\Omega \mid i \in \mathcal{I}^\Omega\}$ be a polyhedral cover of the polygon Ω. First, note that

$$f^{-1}(\Omega) = \bigcup_{i\in\mathcal{I}^\Omega} \{(x,u,w) \in \Upsilon \mid f(x,u,w) \in \mathcal{P}_i^\Omega\} \quad (48a)$$

$$= \bigcup_{(i,j)\in\mathcal{I}^\Omega\times\mathcal{I}^\Upsilon} \Big\{(x,u,w) \in \mathcal{P}_j^\Upsilon \mid A_j x + B_j u + G_j w + g_j \in \mathcal{P}_i^\Omega\Big\}. \quad (48b)$$

Since each nonempty set

$$\{(x,u,w) \in \mathcal{P}_j^\Upsilon \mid A_j x + B_j u + G_j w + g_j \in \mathcal{P}_i^\Omega\}$$

is a polyhedron, it follows that $\Phi := f^{-1}(\Omega)$ is a polygon with a easily-derived polyhedral cover.

Next, recall that the projection of the union of a finite number of sets is the union of the projections of the individual sets, hence the projection of a polygon is also a polygon, i.e., if Σ is a polygon with a polyhedral cover $\mathcal{P}^\Sigma := \{\mathcal{P}_i^\Sigma \mid i \in \mathcal{I}^\Sigma\}$, then $\{\mathrm{Proj}_X \mathcal{P}_i^\Sigma \mid i \in \mathcal{I}^\Sigma\}$ is a polyhedral cover of the polygon $\mathrm{Pre}(\Omega)$. Note also that Υ is a polygon, hence, its projection onto $X \times U$ is a polygon. Similarly, if $\Upsilon \setminus \Phi$ is a polygon, then so too is its projection onto $X \times U$.

What remains to be shown is that $\Upsilon \setminus \Phi$ and Σ are polygons. This follows immediately from referring to Appendix II, where it is shown that the set difference between two polygons is also a polygon. ■

The proof of the following result follows similar arguments as in Theorem 2 and Corollary 1 by noting that the projection of a closed polygon is a closed polygon.

Corollary 3: Suppose **A1)** and **A2)** hold. If the target set X_f is a (closed/compact) polygon (and Υ is a closed/compact polygon), then each set X_i, $i \in \mathbb{N}$, computed as in (20), is a (closed/compact) polygon.

The results above can be combined with the results presented in Section II to develop and implement a number of "first-attempt" algorithms for reachability analysis of piecewise affine systems, based on the prototype algorithms in Sections II-C and E.3. The set differences can be computed using the results in Appendix II and the inverse maps are obtained directly from (48); all other operations, such as projection and Minkowski summation, are standard and relevant software is readily available [31], [42]–[44]. These algorithms can then be analyzed and used as a basis for proposing and comparing more efficient algorithms.

It is important to note that, in practice, different computational geometry problems benefit greatly from modifying an algorithm in subtle, but important ways. A practical algorithm with a meaningful complexity bound can only be obtained by looking at the exact problem structure and choosing the right subset of methods from one or more computational geometry software libraries.

By considering a few special cases, we have provided a number of results that allow for the derivation of different

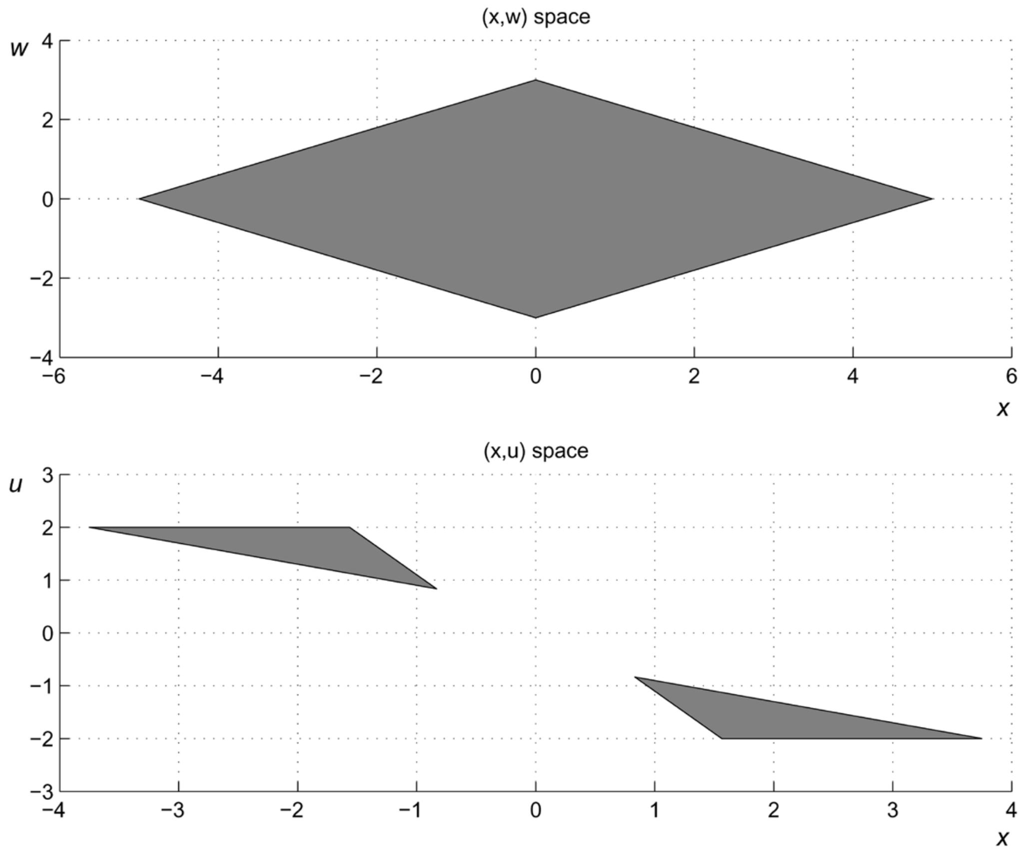

Fig. 4. Graph of (top) $\mathcal{W}$ and (bottom) the set Σ.

algorithms that exploit the system structure. Since there is such a large class of special cases and the various algorithms for polyhedra can be combined in any number of suitable ways, it is beyond the scope of this paper to propose a specific, detailed algorithm and to derive rigorous computational complexity results. An important research topic would be to assume a certain problem structure, use one of the results in this paper, develop the algorithmic details and analyze its complexity when implemented with different computational geometry software libraries.

Remark 10: Clearly, all the results in this section still hold if the system is linear. Once again, there may be many computational and theoretical benefits in exploiting the linearity of the system, convexity of the sets or if the constraints are decoupled. However, it is important to note that, even if all the sets are convex and the system is linear, there is no guarantee that $\mathrm{Pre}(\Omega)$ is convex if the disturbance is dependent on the state and input. This claim is justified in Section IV-A via a numerical example. Note that this is in strong contrast to the well-known fact that $\mathrm{Pre}(\Omega)$ is convex if all the sets are convex and the disturbance is not dependent on the state and input.

IV. Examples

In order to illustrate our results, we consider two simple examples. In the first, the system is scalar and the disturbance state-dependent ($w \in \mathcal{W}(x)$); in the second, the system is second-order and the disturbance control-dependent ($w \in \mathcal{W}(u)$).

A. *Scalar System With State-Dependent Disturbances*

We consider the following scalar system:

$$x^+ = x + u + w \tag{49}$$

which is subject to the constraints $(x,u) \in \mathcal{X} \times \mathcal{U}$ where

$$\mathcal{X} := \{x \mid -5 \le x \le 20\} \text{ and}$$
$$\mathcal{U} := \{u \mid -2 \le u \le 2\}. \tag{50}$$

The state-dependent disturbance satisfies

$$w \in \mathcal{W}(x) \Leftrightarrow (x,w) \in \Delta := \Delta_1 \cup \Delta_2 \tag{51}$$

where

$$\Delta_1 = \mathrm{convh}\{(0,0.25),\ (0,-0.25),\ (2,1.25)$$
$$(2,-1.25),\ (20,2.25),\ (20,-2.25)\}$$

and

$$\Delta_2 = \mathrm{convh}\{(0,0.25),\ (0,-0.25),\ (-2,1.25)$$
$$(-2,-1.25),\ (-20,2.25),\ (-20,-2.25)\}.$$

The set Δ is shown in Fig. 2. The target set is $X_0 = \Omega = \{x \mid -0.6 \le x \le 0.6\}$, which was chosen to be robust control invariant.

The sequence of i-step controllable is computed by using the results of Theorem 1 and some of the sets are: $X_1 = \{x \mid -0.7 \le x \le 0.7\}$, $X_2 = \{x \mid -0.9 \le x \le 0.9\}$, $X_3 = \{x \mid -1.3 \le x \le 1.3\}$, $X_4 = \{x \mid -2.0468 \le x \le 2.0468\}, \ldots, X_8 = \{x \mid -4.5793 \le x \le 4.5793\}$, $X_9 = \{x \mid -5 \le x \le 5.1131\}$, $X_{10} = \{x \mid -5 \le x \le 5.6123\}, \ldots, X_{49} = \{x \mid -5 \le x \le 12.2759\}$, and $X_{50} = \{x \mid -5 \le x \le 12.3099\}$. The set X_∞ of all states

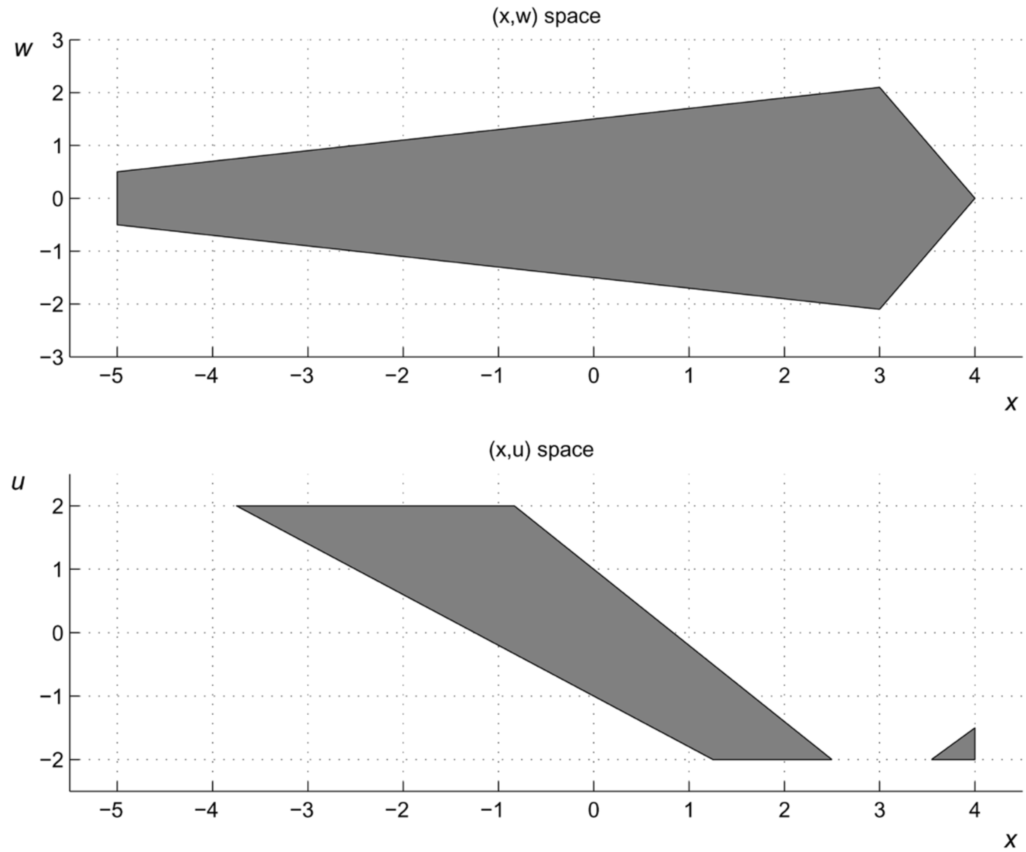

Fig. 5. Graph of (top) $\mathcal{W}$ and (bottom) the set Σ (bottom).

that can be steered to the target set, while satisfying state and control constraints, for all allowable disturbance sequences, is: $X_\infty = \{x \mid -5 \le x \le 12.7999\}$. The sets Σ_i for $i = 1, 2, 3, 4$ are also shown in Fig. 3.

In order to illustrate the fact that the i-step controllable set can be nonconvex even if $\mathcal{X}, \mathcal{U}, \Omega$ and the graph of $\mathcal{W}(x)$ are convex, consider the same example. This time the state-dependent disturbance satisfies $w \in \mathcal{W}(x) \Leftrightarrow (x, w) \in \Delta$ where

$$\Delta := \text{convh}\{(-5, 0), (0, -3), (5, 0), (0, 3)\}. \qquad (52)$$

If the target set is $X_0 = \Omega = \{x \mid -2.5 \le x \le 2.5\}$, the one-step set is $X_1 = \{x \mid -3.75 \le x \le -0.8333\} \cup \{x \mid 0.8333 \le x \le 3.75\}$. The sets Δ and Σ are shown in Fig. 4.

Even if Ω is a robust control invariant set, the convexity of each i-step set still cannot be guaranteed. This is easily illustrated by considering the same example with $\mathcal{X} = \{x \mid -5 \le x \le 4\}$ and $w \in \mathcal{W}(x) \Leftrightarrow (x, w) \in \Delta$ where $\Delta := \text{convh}\{(-5, 0.5), (-5, -0.5), (3, -2.1), (4, 0), (3, 2.1)\}$ and the robust control invariant target set $X_0 = \Omega = \{x \mid -2.5 \le x \le 2.5\}$. In this case, the one-step robust control invariant set is $X_1 = \{x \mid -3.75 \le x \le 2.5\} \cup \{x \mid 3.5455 \le x \le 4\}$. The sets Δ and Σ are shown in Fig. 5.

B. Second-Order LTI Example With Control-Dependent Disturbances

The discrete-time linear time-invariant system

$$x^+ = \begin{bmatrix} 0.7969 & -0.2247 \\ 0.1798 & 0.9767 \end{bmatrix} x + \begin{bmatrix} 0.1271 \\ 0.0132 \end{bmatrix} u + w \qquad (53)$$

is subject to the state and control constraints $(x, u) \in \mathcal{X} \times \mathcal{U}$ with

$$\mathcal{X} := \{x \mid \|x\|_\infty \le 10, [-1\ 1]x \le 12\} \text{ and}$$
$$\mathcal{U} := \{u \mid -3 \le u \le 3\}. \qquad (54)$$

The control-dependent disturbance satisfies

$$w \in \mathcal{W}(u) \Leftrightarrow (u, w) \in \Delta := \Delta_1 \cup \Delta_2 \qquad (55)$$

where Δ_1 and Δ_2 are given by

$$\Delta_1 = \left\{ (u, w) \,\middle|\, \begin{bmatrix} -0.008 & 0 & -1 \\ -1 & 0 & 0 \\ -0.008 & 1 & 0 \\ -0.008 & -1 & 0 \\ -0.008 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ w \end{bmatrix} \le \begin{bmatrix} 0.01 \\ 0 \\ 0.01 \\ 0.01 \\ 0.01 \end{bmatrix} \right\} \qquad (56)$$

and

$$\Delta_2 = \left\{ (u, w) \,\middle|\, \begin{bmatrix} 0.008 & 0 & 1 \\ 1 & 0 & 0 \\ 0.008 & -1 & 0 \\ 0.008 & 1 & 0 \\ 0.008 & 0 & -1 \end{bmatrix} \begin{bmatrix} u \\ w \end{bmatrix} \le \begin{bmatrix} 0.01 \\ 0 \\ 0.01 \\ 0.01 \\ 0.01 \end{bmatrix} \right\}. \qquad (57)$$

The target set is

$$X_0 = \text{convh}\{(-0.2035, 0.0482),\ (0.2035, -0.0482)$$
$$(-0.2035, -0.0148),\ (-0.1405, 0.0482)$$
$$(0.2035, 0.0148),\ (0.1405, -0.0482)\}$$

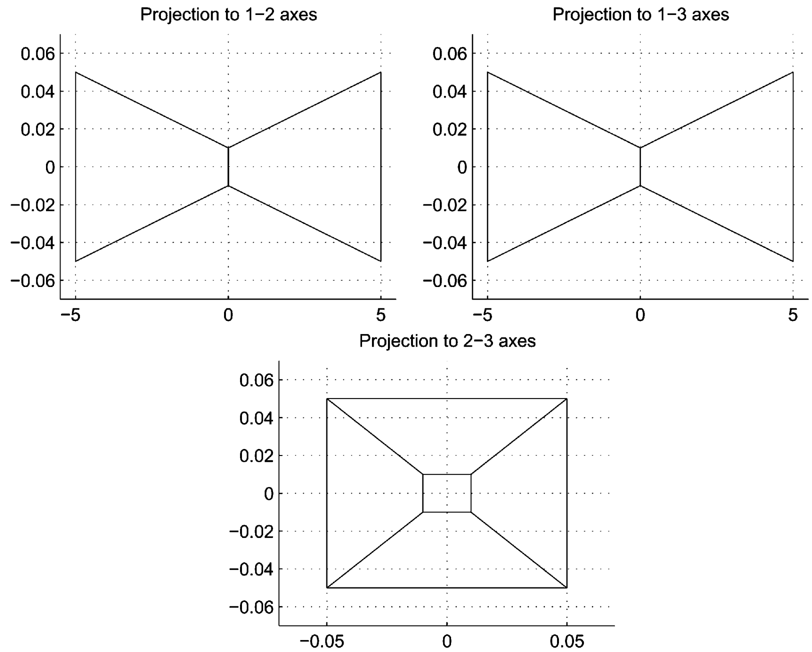

Fig. 6. Graph of $\mathcal{W}$.

which was chosen to be robust control invariant. The projections of the set Δ onto two-dimensional subspaces are shown in Fig. 6. Some of the i-step controllable sets, computed using Theorem 1, are shown in Fig. 7.

V. Conclusion

The main result of this paper (Theorem 1) showed how one can obtain $\text{Pre}(\Omega)$, the set of states that can be robustly steered to Ω, via the computation of a sequence of set differences and projections. It was then shown in Theorem 4 that if Ω and the relevant constraint sets are polygons (i.e., they are given by the unions of finite sets of convex polyhedra) and the system is linear or piecewise affine, then $\text{Pre}(\Omega)$ is also a polygon and can be computed using standard computational geometry software. In particular, new results were given in Appendix II which allow one to compute the set difference for (possibly nonconvex) polygons by solving a finite number of LPs. Finally, some simple examples were given which show that, even if the system is linear, the respective constraint sets are convex and the target set is robust control invariant, convexity of the i-step controllable sets cannot be guaranteed.

Future work could focus on using the results in this paper to develop efficient algorithms that exploit system structure. For piecewise affine systems, the complexity of the description of the output of a reachability computation might, in general, be worst-case exponential in terms of the size of the input data. Clearly, there is nothing that one could do about the inherent complexity of a solution, except maybe through making suitable approximations during computation time. However, as is common practice in computational geometry [30], [31], it may be more appropriate to analyze the complexity of a reachability algorithm not only in terms of the size of the input data, but also in terms of the size of the output data. In computational geometry, an algorithm is said to be tractable if it has a computational complexity that is a polynomial function of the size of the input *and* output data. This notion could also be applied to the rigorous analysis of the complexity of reachability algorithms for piecewise affine systems.

Appendix I
Results on Set-Valued Functions

The definitions of inner and outer semicontinuity employed below are due to [47]; for Definitions 1–4 and Theorem 5, see [48]; Polak also provided the proof of Proposition 2 (private communication). In what follows, $B(z, \rho) := \{z \mid \|z\| \leq \rho\}$ and $d(a, A) := \inf_{b \in A} \|a - b\|$.

Definition 2: A set-valued map $F : \text{IR}^r \rightarrow 2^{\text{IR}^n}$ is outer semi-continuous (o.s.c.) at $\hat{z}$ if $F(\hat{z})$ is closed and, for every compact set S such that $F(\hat{z}) \cap S = \emptyset$, there exists a $\rho > 0$ such that $F(z) \cap S = \emptyset$ for all $z \in B(\hat{z}, \rho)$. A set-valued map $F : \text{IR}^r \rightarrow 2^{\text{IR}^n}$ is o.s.c. if it is o.s.c. at every $z \in \text{IR}^r$.

Definition 3: A set-valued map $F : \text{IR}^r \rightarrow 2^{\text{IR}^n}$ is inner semi-continuous (i.s.c.) at $\hat{z}$ if $F(\hat{z})$ is closed and, for every open set S such that $F(\hat{z}) \cap S \neq \emptyset$, there exists a $\rho > 0$ such that $F(z) \cap S \neq \emptyset$ for all $z \in B(\hat{z}, \rho)$. A set-valued map $F : \text{IR}^r \rightarrow 2^{\text{IR}^n}$ is i.s.c. if it is i.s.c. at every $z \in \text{IR}^r$.

Definition 4: A set-valued map $F : \text{IR}^r \rightarrow 2^{\text{IR}^n}$ is continuous if it is both o.s.c. and i.s.c.

Definition 5: A point $\hat{a}$ is a limit point of the infinite sequence of sets $\{A_i\}$ if $d(\hat{a}, A_i) \rightarrow 0$. A point $\hat{a}$ is a cluster point if there exists a subsequence $I \subset N$ such that $d(\hat{a}, A_i) \rightarrow 0$ as $i \rightarrow \infty$, $i \in I$. The set $\limsup A_i$ is the set of cluster points of $\{A_i\}$ and $\liminf A_i$ is the set of limit points of $\{A_i\}$, i.e., $\limsup A_i$ is the set of cluster points of sequences $\{a_i\}$ such that $a_i \in A_i$ for all $i \in \text{IN}$ and $\liminf A_i$ is the set of limits of sequences $\{a_i\}$ such that $a_i \in A_i$ for all $i \in \text{IN}$. The sets A_i converge to the set A ($A_i \rightarrow A$ or $\lim A_i = A$) if $\limsup A_i = \liminf A_i = A$.

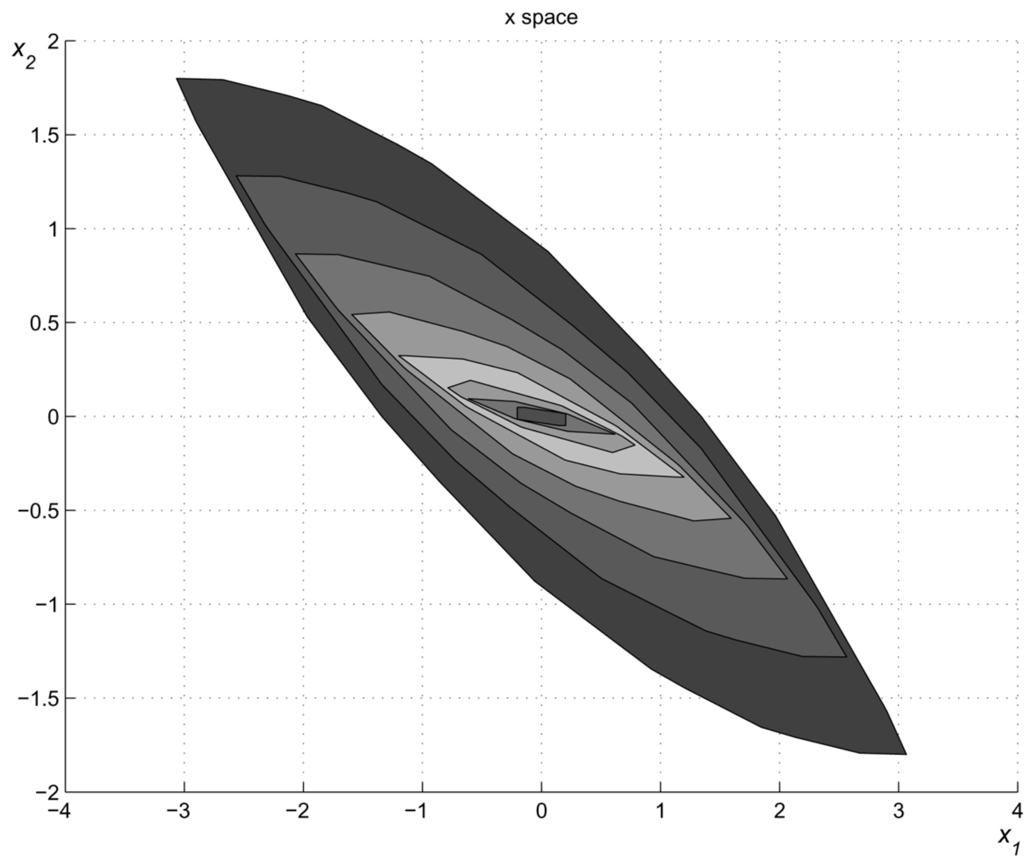

Fig. 7. Sets X_i for $i = 0, 1, \ldots, 7$.

The following result appears as [48, Th. 5.3.7].

Theorem 5:

i) A function $F : \mathrm{IR}^r \to 2^{\mathrm{IR}^n}$ is o.s.c. at $\hat{z}$ if and only if for any sequence $\{z_i\}$ such that $z_i \to \hat{z}$, $\limsup F(z_i) \subseteq F(\hat{z})$. Also, F is o.s.c. if and only if it graph $G := \{(z, y) \mid y \in F(z)\}$ is closed.

ii) A function $F : \mathrm{IR}^r \to 2^{\mathrm{IR}^n}$ is i.s.c. at $\hat{z}$ if and only if for any sequence $\{z_i\}$ such that $z_i \to \hat{z}$, $\liminf F(z_i) \supseteq F(\hat{z})$.

iii) Suppose $F : \mathrm{IR}^r \to 2^{\mathrm{IR}^n}$ is such that $F(z)$ is compact for all $z \in \mathrm{IR}^r$ and bounded on bounded sets. Then, F is o.s.c. at $\hat{z}$ if and only if, for every open set S such that $F(\hat{z}) \subseteq S$, there exists a $\rho > 0$ such that $F(z) \subseteq S$ for all $z \in B(\hat{z}, \rho)$.

Proposition 2: Suppose that $f : \mathrm{IR}^r \times \mathrm{IR}^p \to \mathrm{IR}^n$ is continuous and that $\mathcal{W} : \mathrm{IR}^p \to 2^{\mathrm{IR}^p}$ is continuous and bounded on bounded sets. Then the set-valued function $F : \mathrm{IR}^r \to 2^{\mathrm{IR}^n}$ defined by $F(z) := \{f(z, w) \mid w \in \mathcal{W}(z)\}$ is continuous.

Proof:

i) (F is o.s.c.). Let $\{z_i\}$ be any infinite sequence such that $z_i \to \hat{z}$ and let $\{f_i\}$ be any infinite sequence such that $f_i \in F(z_i)$ for all $i \in \mathrm{IN}$ and $f_i \to \hat{f}$. Then, for all i, $f_i = f(z_i, w_i)$ with $w_i \in \mathcal{W}(z_i)$. Since $\{z_i\}$ lies in a compact set and $\mathcal{W} : \mathrm{IR}^p \to 2^{\mathrm{IR}^p}$ is bounded on bounded sets, there exists a subsequence of $\{w_i\}$ such that $w_i \to \hat{w}$ as $i \to \infty$, $i \in I \subset \mathrm{IN}$. Since $\mathcal{W}$ is continuous, $\hat{w} \in \mathcal{W}(\hat{z})$. Hence

$$\hat{f} = \lim_{i \in I} f(z_i, w_i) = f(\hat{z}, \hat{w}) \in F(\hat{z}).$$

This implies that F is o.s.c.

ii) (F is i.s.c.) Let $\{z_i\}$ be any infinite sequence such that $z_i \to \hat{z}$ and let $\hat{f}$ be an arbitrary point in $F(\hat{z})$. Then, $\hat{f} = f(\hat{z}, \hat{w})$ for some $\hat{w} \in \mathcal{W}(\hat{z})$. Since $\mathcal{W}$ is continuous, there exists an infinite sequence $\{w_i\}$ such that $w_i \in \mathcal{W}(z_i)$ and $w_i \to \hat{w}$. Then $f_i := f(z_i, w_i) \in F(z_i)$ for all $i \in \mathrm{IN}$ and

$$\lim_{i \to \infty} f_i = \lim_{i \to \infty} f(z_i, w_i) = f(\hat{z}, \hat{w}) = \hat{f} \in F(\hat{z}).$$

This implies that F is i.s.c.

■

Proposition 3: Suppose $F : \mathrm{IR}^r \to 2^{\mathrm{IR}^n}$ is continuous and that $\Omega \subseteq \mathrm{IR}^n$ is closed. Then, the (outer) inverse set $F^{\dagger}(\Omega) := \{z \mid F(z) \subseteq \Omega\}$ is closed.

Proof: Suppose $\{z_i\}$ is an arbitrary infinite sequence in $F^{\dagger}(\Omega)$ ($F(z_i) \subseteq \Omega$ for all $i \in \mathrm{IN}$) such that $z_i \to \hat{z}$. Since F is continuous, $\lim_{i \to \infty} F(z_i) = F(\hat{z})$. Because Ω is closed, $F(z_i) \subseteq \Omega$ for all $i \in \mathrm{IN}$ implies $F(\hat{z}) \subseteq \Omega$. Hence, $\hat{z} \in F^{\dagger}(\Omega)$ so that $F^{\dagger}(\Omega)$ is closed. ■

APPENDIX II
SET DIFFERENCE BETWEEN POLYGONS

Since we were unable to find any specific details in the literature on computing the set difference and/or complement of polygons, we include some basic results in this section that are easily implemented using standard computational geometry software libraries. Further research could be focused on deriving more efficient algorithms that exploit any structure in the problem.

Before proceeding, let $\mathrm{IN}_n := \{1, 2, \ldots, n\}$ and recall that the complement of A in IR^n is $A^c := \mathrm{IR}^n \setminus A$ and that $B \setminus A = B \cap A^c$. In other words, the set difference operation also allows us to compute the complement of a set in a given space. For the definitions of a polyhedron, polygon and polyhedral cover, see Section III.

The first result, which is due to [45], allows one to compute the set difference between two polyhedra.

Proposition 4: Let $A \subset \mathrm{IR}^n$ be a polyhedron and let $B := \{x \in \mathrm{IR}^n \mid c_i'x \leq d_i, i = 1, \ldots, r\}$ be a nonempty polyhedron, where all the $c_i \in \mathrm{IR}^n$ and $d_i \in \mathrm{IR}$. If

$$S_1 := \{x \in A \mid c_1'x > d_1\} \tag{58a}$$

$$S_i := \left\{x \in A \mid c_i'x > d_i,\ c_j'x \leq d_j \qquad \forall\, j \in \mathrm{IN}_{i-1}\right\},\quad i = 2, \ldots, r, \tag{58b}$$

then the set difference $A \setminus B = \bigcup_{i=1}^{r} S_i$ is a polygon. Furthermore, the polyhedral cover $\{S_i \neq \emptyset \mid i \in \mathrm{IN}_r\}$ is a partition of $A \setminus B$.

Remark 11: Note that in order to simplify notation B was assumed to be closed, whereas no similar assumption on A has been made. Clearly, Proposition 4 is without loss of generality, since the result is trivially extended (at the expense of an increase in notational complexity) for the case when B is not closed.

In practice, computation time can be reduced by checking whether $A \cap B$ is empty or whether $A \subseteq B$ before actually computing $A \setminus B$; if $A \cap B = \emptyset$, then $A \setminus B = A$ and if $A \subseteq B$, then $A \setminus B = \emptyset$. Using an extended version of Farkas' Lemma [7, Lemma 4.1], checking whether one polyhedron is contained in another amounts to solving a single LP. Alternatively, one can solve a finite number of smaller LPs to test for set inclusion [32, Prop. 3.4].

Once $A \setminus B$ has been computed, the memory requirements can be reduced by removing all empty S_i and removing any redundant inequalities describing the nonempty S_i. Checking whether a polyhedron is nonempty can be done by solving a single linear program (LP). Removing redundant inequalities can be done by solving a finite number of LPs [30]. As a general rule, it is usually a good idea to determine first whether a polyhedron is nonempty or not before removing redundant inequalities.

The second result allows one to compute the set difference between a polygon and a polyhedron.

Proposition 5: Let $\left\{\mathcal{P}_j^C \mid j \in \mathrm{IN}_p\right\}$ be a polyhedral cover of the polygon C. If A is a nonempty polyhedron, then each nonempty $\mathcal{P}_j^C \setminus A$, $j \in \mathrm{IN}_p$, is a polygon and the set difference

$$C \setminus A = \bigcup_{j=1}^{p} \left(\mathcal{P}_j^C \setminus A\right) \tag{59}$$

is also a polygon.

Proof: This follows trivially from $C \setminus A = \left(\bigcup_{j=1}^{p} \mathcal{P}_j^C\right) \cap A^c = \bigcup_{j=1}^{p} \left(\mathcal{P}_j^C \cap A^c\right)$. ■

Note that if the polyhedral cover $\left\{\mathcal{P}_j^C \mid j \in \mathrm{IN}_p\right\}$ is a partition of C and $C \setminus A$ is nonempty, then it is easy to compute a polyhedral cover of $C \setminus A$ that is also a partition of $C \setminus A$, provided Proposition 4 were used to compute the polyhedral cover of each $\mathcal{P}_j^C \setminus A$, $j \in \mathrm{IN}_p$.

The last result allows one to compute the set difference between two polygons.

Proposition 6: Let $\left\{\mathcal{P}_j^C \mid j \in \mathrm{IN}_p\right\}$ and $\left\{\mathcal{P}_k^D \mid k \in \mathrm{IN}_q\right\}$ be polyhedral covers of the polygons C and D, respectively. If

$$E_0 := C, \tag{60a}$$

$$E_k := E_{k-1} \setminus \mathcal{P}_k^D \qquad \forall\, k \in \mathrm{IN}_q \tag{60b}$$

then the set difference $C \setminus D = E_q$ is a polygon.

Proof: The result follows from noting that

$$\begin{aligned}
C \setminus D &= C \cap D^c && \text{(61a)}\\
&= C \cap \left(\cup_{k=1}^{q} \mathcal{P}_k^D\right)^c = C \cap \left(\cap_{k=1}^{q} \left(\mathcal{P}_k^D\right)^c\right) && \text{(61b)}\\
&= C \cap \left(\mathcal{P}_1^D\right)^c \cap \left(\mathcal{P}_2^D\right)^c \cap \cdots \cap \left(\mathcal{P}_q^D\right)^c && \text{(61c)}\\
&= \left(C \cap \left(\mathcal{P}_1^D\right)^c\right) \cap \left(\mathcal{P}_2^D\right)^c \cap \cdots \cap \left(\mathcal{P}_q^D\right)^c && \text{(61d)}\\
&= \left(C \setminus \mathcal{P}_1^D\right) \cap \left(\mathcal{P}_2^D\right)^c \cap \cdots \cap \left(\mathcal{P}_q^D\right)^c && \text{(61e)}\\
&= \left(\left(C \setminus \mathcal{P}_1^D\right) \setminus \mathcal{P}_2^D\right) \cap \cdots \cap \left(\mathcal{P}_q^D\right)^c && \text{(61f)}\\
&= \left(\cdots \left(\left(C \setminus \mathcal{P}_1^D\right) \setminus \mathcal{P}_2^D\right) \setminus \cdots\right) \setminus \mathcal{P}_q^D. && \text{(61g)}
\end{aligned}$$

Letting $E_0 := C$ and $E_k := E_{k-1} \setminus \mathcal{P}_k^D$, $\forall\, k \in \mathrm{IN}_q$, yields the claim. ■

Clearly, polyhedral covers for all the polygons $E_{k-1} \setminus \mathcal{P}_k^D$, $k \in \mathrm{IN}_q$, can be computed using Proposition 5. Note also that if the polyhedral cover $\left\{\mathcal{P}_j^C \mid j \in \mathrm{IN}_p\right\}$ is a partition of C and $C \setminus D$ is nonempty, then it is easy to compute a polyhedral cover of E_q that is also a partition of $C \setminus D$ by using Propositions 4 and 5 to compute polyhedral covers for all the E_k, $k \in \mathrm{IN}_q$.

ACKNOWLEDGMENT

The authors would like to thank Prof. E. Polak for helpful discussions. All computations were performed with the Geometric Bounding Toolbox [43].

REFERENCES

[1] C. A. Desoer and J. Wing, "A minimal time discrete system," *IRE Trans. Autom. Control*, pp. 111–125, May 1961.

[2] H. S. Witsenhausen, "A minimax control problem for sampled linear systems," *IEEE Trans. Autom. Control*, vol. AC-13, no. 1, pp. 5–21, Feb. 1968.

[3] M. C. Delfour and S. K. Mitter, "Reachability of perturbed systems and min sup problems," *SIAM J. Control*, vol. 7, no. 4, pp. 521–533, Nov. 1969.

[4] D. P. Bertsekas and I. B. Rhodes, "On the minimax reachability of target sets and target tubes," *Automatica*, vol. 7, pp. 233–247, 1971.

[5] J. D. Glover and F. C. Schweppe, "Control of linear dynamic systems with set constrained disturbances," *IEEE Trans. Autom. Control*, vol. AC-16, pp. 411–423, Oct. 1971.

[6] D. P. Bertsekas, "Infinite-time reachability of state-space regions by using feedback control," *IEEE Trans. Autom. Control*, vol. AC-17, no. 5, pp. 604–613, May 1972.

[7] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[8] J.-P. Aubin, J. Lygeros, M. Quincampoix, S. Sastry, and N. Seube, "Impulse differential inclusions: A viability approach to hybrid systems," *IEEE Trans. Autom. Control*, vol. 47, no. 1, pp. 2–20, Jan. 2002.

[9] E. De Santis, M. D. Di Benedetto, and L. Berardi, "Computation of maximal safe sets for switching systems," *IEEE Trans. Autom. Control*, vol. 49, no. 2, pp. 184–195, Feb. 2004.

[10] L. C. G. J. M. Habets and J. H. van Schuppen, "A controllability result for piecewise linear hybrid systems," in *Proc. 6th Eur. Control Conf.*, Porto, Portugal, Sep. 2001, pp. 3870–3873.

[11] J. Lygeros, C. Tomlin, and S. Sastry, "Controllers for reachability specifications for hybrid systems," *Automatica*, vol. 35, no. 3, pp. 349–70, Mar. 1999.

[12] C. Tomlin, I. Mitchell, A. Bayen, and M. Oishi, "Computational techniques for the verification and control of hybrid systems," *Proc. IEEE*, vol. 91, no. 7, pp. 986–1001, Jul. 2003.

[13] F. Torrisi, "Modeling and reach-set computation for analysis and optimal control of discrete hybrid automata," Ph.D. dissertation, ETH, Zürich, Switzerland, 2003.

[14] X. D. Koutsoukos and P. J. Antsaklis, "Safety and reachability of piecewise linear hybrid dynamical systems based on discrete abstractions," *J. Discrete Event Dyna. Syst.*, vol. 13, no. 3, pp. 203–243, 2003.

[15] R. Vidal, S. Schaffert, O. Shakernia, J. Lygeros, and S. Sastry, "Decidable and semi-decidable controller synthesis for classes of discrete time hybrid systems," in *Proc. 40th IEEE Conf. Decision and Control*, Orlando, FL, Dec. 2001, pp. 1243–1248.

[16] F. Blanchini, "Ultimate boundedness control for uncertain discrete-time systems via set-induced Lyapunov functions," *IEEE Trans. Autom. Control*, vol. 39, no. 2, pp. 428–433, Feb. 1994.

[17] E. De Santis, "Invariant sets: A generalization to constrained systems with state dependent disturbances," in *Proc. 37th IEEE Conf. Decision and Control*, Tampa, FL, Dec. 1998, pp. 622–623.

[18] M. Jirstrand, "Nonlinear control system design by quantifier elimination," *J. Symb. Comput.*, vol. 24, no. 2, pp. 137–152, 1997.

[19] R. Ghosh and C. Tomlin, "Symbolic reachable set computation of piecewise affine hybrid automata and its application to biological modeling: Delta-notch protein signalling," *Syst. Biol.*, vol. 1, no. 1, pp. 170–183, Jun. 2004.

[20] G. Lafferriere, G. J. Pappas, and S. Yovine, "Symbolic reachability computation for families of linear vector fields," *J. Symb. Comput.*, vol. 32, no. 3, pp. 231–253, Sep. 2001.

[21] A. Dolzmann and T. Sturm, "REDLOG: Computer algebra meets computer logic," Universitä Passau, Fakultät Für Mathematik und Informati, Passau, Germany, Tech. Rep. MIP-9603, 1996.

[22] H. Yanami and H. Anai, "Development of synrac—formula description and new functions," in *Proc. Int. Workshop on Computer Algebra Systems and their Applications (CASA) 2004: ICCS 2004*, vol. LNCS 3039, pp. 286–294.

[23] H. Anai and V. Weispfenning, "Reach set computation using real quantifier elimination," in *Proc. Int. Workshop on Hybrid Systems: Computation and Control (HSCC2001)*, vol. 2034, 2001, pp. 63–76.

[24] R. Loos and V. Weispfenning, "Applying linear quantifier elimination," *Comput. J.*, vol. 36, no. 5, pp. 450–462, 1993.

[25] B. F. Caviness and J. R. Johnson, Eds., *Quantifier Elimination and Cylindrical Algebraic Decomposition*. ser. Texts and mographs in symbolic computation. New York: Springer-Verlag, 1998.

[26] V. Weispfenning, "Simulation and optimization by quantifier elimination," *J. Symb. Comput.*, vol. 24, no. 2, pp. 189–208, 1997.

[27] B. Anderson, N. Bose, and E. Jury, "Output feedback stabilization and related problems-solution via decision methods," *IEEE Trans. Autom. Control*, vol. AC-20, no. 1, pp. 53–65, Jan. 1975.

[28] H. Anai and S. Hara, "Fixed-structure robust controller synthesis based on sign definite condition by a special quantifier elimination," in *Proc. Amer. Control Conf.*, 2000, pp. 1312–1316.

[29] D. Q. Mayne, "Control of constrained dynamic systems," *Eur. J. Control*, vol. 7, pp. 87–99, 2001.

[30] K. Fukuda. (2004) Frequently asked questions in polyhedral computation. [Online]. Available: http://www.ifor.math.ethz.ch/~fukuda/fukuda.html

[31] C. N. Jones, E. C. Kerrigan, and J. M. Maciejowski. (2004) Equality set projection: A new algorithm for the projection of polytopes in halfspace representation. Dept. Eng., Univ. Cambridge, Cambridge, U.K.. [Online]. Available: http://www-control.eng.cam.ac.uk

[32] E. C. Kerrigan, "Robust constraint satisfaction: Invariant sets and predictive control," Ph.D. dissertation, Dept. Eng., Univ. Cambridge, Cambridge, U.K., 2000.

[33] E. C. Kerrigan, J. Lygeros, and J. M. Maciejowski, "A geometric approach to reachability computations for constrained discrete-time systems," in *Proc. 15th IFAC World Congr. Automatic Control*, Barcelona, Spain, Jul. 2002.

[34] E. C. Kerrigan and D. Q. Mayne, "Optimal control of constrained, piecewise affine systems with bounded disturbances," in *Proc. 41st IEEE Conf. Decision and Control*, Las Vegas, NV, Dec. 2002, pp. 1552–1557.

[35] S. V. Raković, E. C. Kerrigan, and D. Q. Mayne, "Reachability computations for constrained discrete-time systems with state- and input-dependent disturbances," in *Proc. 42nd IEEE Conf. Decision and Control*, Maui, HI, Dec. 2003, pp. 3905–3910.

[36] S. V. Raković, "Robust control of constrained discrete time systems: Characterization and implementation," Ph.D. dissertation, Imperial College London, London, U.K., 2005.

[37] S. V. Raković, E. C. Kerrigan, and D. Q. Mayne, "Optimal control of constrained piecewise affine systems with state- and input-dependent disturbances," in *Proc. 16th Int. Symp. Mathematical Theory of Networks and Systems*, Brussels, Belgium, Jul. 2004.

[38] S. Rakovic, P. Grieder, M. Kvasnica, D. Q. Mayne, and M. Morari, "Computation of invariant sets for piecewise affine discrete time systems subject to bounded disturbances," in *Proc. 43rd IEEE Conf. Decision and Control*, Paradise Island, Bahamas, Dec. 2004, pp. 1418–1423.

[39] I. Kolmanovsky and E. G. Gilbert, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Math. Prob. Eng.*, vol. 4, no. 4, pp. 317–367, 1998.

[40] G. Matheron, *Random Sets and Integral Geometry*. New York: Wiley, 1975.

[41] J. Serra, *Image Analysis and Mathematical Morphology*. New York: Academic, 1982.

[42] K. Fukuda. cdd/cdd+ Reference Manual. [Online]. Available: http://www.ifor.math.ethz.ch/~fukuda/fukuda.html

[43] S. M. Veres. Geometric Bounding Toolbox (GBT 7.2) for MATLAB. [Online]. Available: http://sysbrain.com

[44] M. Kvasnica, P. Grieder, M. Baotić, and M. Morari, "Multi parametric toolbox (MPT)," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science. Philadelphia, PA: Springer-Verlag, 2004, vol. 2993, pp. 448–462.

[45] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, pp. 3–20, 2002.

[46] B. Grünbaum, *Convex Polytopes*, 2nd ed ed. New York: Springer-Verlag, 2003.

[47] R. T. Rockafellar and R. J.-B Wets, *Variational Analysis*. New York: Springer-Verlag, 1998.

[48] E. Polak, *Optimization: Algorithms and Consistent Approximations*. New York: Springer-Verlag, 1997.

Saša V. Raković received the B.Sc. degree in electrical engineering from the Technical Faculty Cacak, University of Kragujevac, Serbia and Montenegro, in 1999, and the M.Sc. degree in control engineering and the Ph.D. degree in control theory from Imperial College London, U.K., in 2001 and 2005, respectively.

He is currently a Research Associate in the Control and Power Research Group at Imperial College. His research interests include set invariance theory, reachability analysis and robust model predictive control, and optimization-based design.

Eric Kerrigan (S'94–M'02) received the B.Sc.(Eng) degree in electrical engineering from the University of Cape Town, South Africa, in 1996, and the Ph.D. degree from the University of Cambridge, Cambridge, U.K., in 2001.

During 1997, he was with the Council for Scientific Industrial Research, South Africa. From 2001 to 2005, he was a Research Fellow at the Department of Engineering, University of Cambridge. He is currently a Lecturer and Royal Academy of Engineering Research Fellow at Imperial College London, U.K. His research interests include optimal and robust control of constrained systems, predictive control, optimization, and applications in aeronautics.

David Q. Mayne (S'75–SM'77–F'81) received the Ph.D. and D.Sc. degrees from the University of London, London, U.K., and the degree of Doctor of Technology (*honoris causa*) from the University of Lund, Lund, Sweden.

He has held posts at the University of the Witwatersrand, the British Thomson Houston Company, University of California, Davis, and Imperial College London, where he is now a Senior Research Fellow. His research interests include optimization, optimization-based design, nonlinear control, and model predictive control.

John Lygeros (S'89–M'97) received the B.Eng. degree in electrical engineering and the M.Sc. degree in control, both from Imperial College London, U.K., in 1990 and 1991, respectively, and the Ph.D. degree from the Electrical Engineering and Computer Sciences Department, University of California, Berkeley, in 1996.

He held a series of postdoctoral research appointments at the National Automated Highway Systems Consortium, MIT, and UC Berkeley. In parallel, he also worked as a part time Research Engineer at SRI International, and as a Visiting Professor at the Mathematics Department of the Universite de Bretagne Occidentale, France. Between July 2000 and March 2003, he was a University Lecturer at the Department of Engineering, University of Cambridge, Cambridge, U.K. Since March 2003, he has been an Assistant Professor at the Department of Electrical and Computer Engineering, University of Patras, Patras, Greece. His research interests include modeling, analysis and control of hierarchical, and hybrid systems, with applications to biochemical networks and large-scale systems such as automated highways and air traffic management.