ON THE COMPUTATION OF VIABLE POLYTOPES FOR LINEAR SYSTEMS

Yan Gao[†], **John Lygeros**[‡]

[†]School of Management
University of Shanghai for Science and Technology
Shanghai 200093, China
email:gaoyan1962@263.net

[‡] Department of Electrical and Computer Engineering
University of Patras
Rio, Patras, GR 26500, Greece
email: lygeros@ee.upatras.gr

Keywords: Linear system, invariance, polytope, optimization.

Abstract

We develop a procedure for computing viable (also known as controlled invariant) polytopes of a given subset of the state space under linear dynamics. The advantage of the proposed algorithm is that at every step it maintains a polytope that is itself viable. Therefore, even if the algorithm is stopped before termination it will still return a viable polytope, that can be subsequently used for controller design.

1 Introduction

The problems of reachability, invariance and viability (controlled invariance) have been extensively studied in the literature for over three decades. Most 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. 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 include both continuous (i.e. real-valued) and discrete (i.e. finite-valued) components. In recent years, invariance and reachability problems for classes of hybrid systems have been studied by a number of authors [7, 10, 6, 9, 15].
When faced with a set that is not viable/invariant, one would like to establish a subset of this set that is viable/invariant (ideally the maximal subset). Most of the algorithms that have been proposed for computing viable and invariant subsets rely on dynamic programming. The algorithms typically start with the whole of the given set and trim away parts that cause viability/invariance problems. If the algorithm terminates the resulting set will typically be the maximal viable/invariant subset of the given set.
One drawback of the dynamic programming approach is that the set maintained by the algorithm is not viable/invariant until the algorithm terminates. This means that if we are forced to stop the computation before termination (due to timing constraints, or because the problem is undecidable and the algorithm would never terminate) the set produced by the algorithm is practically useless for controller design.
In this paper we develop an algorithm for computing conservative approximations for viable sets. The main advantage of the proposed approach is that the set maintained by the algorithm is always viable. Therefore, if the algorithm is stopped before termination it can be used for the design of a safe (albeit conservative) controller. We restrict our attention to the computation of viable polytopic subsets of a given polytope under linear dynamics. This class of problems has been studied by a number of authors [4, 5, 8, 11, 14]. Even though sufficient conditions for viability and invariance can be found in these references, the construction of viable polytopes is not considered.

2 Viable sets and viability kernels

2.1 Background definitions

We start with a brief overview of some standard definitions from viability theory and non-smooth analysis; for a more thorough treatment the reader is referred to [1, 3, 2, 12, 13].
For an arbitrary set K, 2^K denotes the power set of K, i.e. the set of all subsets of K. For a subset of Euclidean space

$W \subseteq \mathbb{R}^n$, we denote by $co(W)$ the convex hull of W, and by $cone(W)$ the cone generated by W, i.e. $cone(W) = \{\lambda w \mid w \in W, \lambda \geq 0\}$.
Consider a differential inclusion of the form

$$\dot{x} \in F(x)$$

where $F : \mathbb{R}^n \to 2^{\mathbb{R}^n}$. A *solution* to the differential inclusion over an interval $[0, T]$ starting at $x_0 \in X$ is an absolutely continuous function $x : [0, T] \to X$, such that $x(0) = x_0$ and almost everywhere $\dot{x}(t) \in F(x(t))$.

Definition 1 (Viable Set) *A solution $x(\cdot) : [0, T] \to \mathbb{R}^n$ of the differential inclusion $\dot{x} \in F(x)$ is called viable in a set $K \subseteq \mathbb{R}^n$, if $x(t) \in K$ for all $t \in [0, T]$. A set $K \subseteq \mathbb{R}^n$ is called locally viable under a differential inclusion $\dot{x} \in F(x)$ if for all $x_0 \in \mathbb{R}^n$ there exists $T > 0$ and a solution $x(\cdot) : [0, T] \to \mathbb{R}^n$ of the differential inclusion with $x(0) = x_0$ that is viable in K. K is called viable if we can take $T = \infty$ in the above.*

For a closed subset, $K \subseteq \mathbb{R}^n$, and a point $x \in K$, we use $T_K(x)$ to denote the *contingent cone* to K at x, i.e. the set of $v \in \mathbb{R}^n$ such that there exists a sequence of real numbers $h_n > 0$ converging to 0 and a sequence of $v_n \in \mathbb{R}^n$ converging to v satisfying

$$\forall\, n \geq 0, \;\; x + h_n v_n \;\in\; K.$$

Notice that the set $T_K(x)$ is a cone; if $y \in T_K(x)$, then $ty \in T_K(x)$ for all $t \geq 0$. Moreover, if x is in the interior of K, $T_K(x) = \mathbb{R}^n$. Finally, if $K_1 \subseteq K_2$ then $T_{K_1}(x) \subseteq T_{K_2}(x)$ for all $x \in K_1$.
We say that $F : \mathbb{R}^n \to 2^{\mathbb{R}^n}$ is *Marchaud* if and only if

1. the graph and the domain of F are nonempty and closed;
2. for all $x \in X$, $F(x)$ is convex, compact and nonempty;
3. the growth of F is linear, that is there exists $c > 0$ such that for all $x \in X$

$$\sup\{|v| \mid v \in F(x)\} \;\leq\; c(|x| + 1).$$

The following characterization of viable sets can be found in [1].

Theorem 1 *Assume F is Marchaud. A closed set $K \subseteq \mathbb{R}^n$ is viable under the differential inclusion $\dot{x} \in F(x)$ if and only if for all $x \in K$, $T_K(x) \cap F(x) \neq \emptyset$.*

Definition 2 (Viability Kernel) *The viability kernel, $Viab_F(K)$ of a set $K \subseteq \mathbb{R}^n$ under a differential inclusion $\dot{x} \in F(x)$ is the set of states $x_0 \in K$ for which there exists an infinite solution $x(\cdot) : [0, \infty) \to \mathbb{R}^n$ of the differential inclusion with $x(0) = x_0$ that is viable in K.*

The following characterization of the viability kernel can be found in [1].

Theorem 2 *Assume F is Marchaud and K is closed. $Viab_F(K)$ is the largest closed subset of K (possibly the empty set) that satisfies the conditions of Theorem 1.*

Finally, the following result can be found in [2].

Theorem 3 *If F is Marchaud and W is a non-empty, compact, convex set which is viable under the differential inclusion $\dot{x} \in F(x)$, then there exists $x \in W$ such that $0 \in F(x)$.*

2.2 Linear systems

Let us now restrict our attention to the case where the dynamics are linear

$$\dot{x} = Ax + Bu, \; x \in K, u \in U. \tag{1}$$

We assume that $K \subseteq \mathbb{R}^n$ and $U \subseteq \mathbb{R}^m$ are closed and convex and that $u(\cdot)$ is measurable as a function of time. We define $F : \mathbb{R}^n \to 2^{\mathbb{R}^n}$ by

$$F(x) = \{v \in \mathbb{R}^n \mid \exists u \in U, \; v = Ax + Bu\}.$$

It is easy to see that $F(\cdot)$ is Marchaud.

Proposition 1 *If $D \subseteq \mathbb{R}^n$ is a viable set of (1), then the convex hull, $co(D)$, of D is also viable. Moreover, the viability kernel, $Viab_F(K)$, is convex.*

Proof: Recall that the solutions of the differential inclusion (1) have the form

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau,$$

where $x(0) = x_0$ and $u(\cdot) : [0, \infty) \to U$ is measurable function. A point $x_0 \in co(D)$ can be written as $x_0 = \sum_{i=1}^{n+1} \lambda_i x_i$, for some $x_i \in D$, $\lambda_i \geq 0$, $i = 1, \ldots, n+1$ with $\sum_{i=1}^{n+1} \lambda_i = 1$. Since D is viable, there exist $u_i(\cdot) : [0, \infty) \to U$ such that

$$e^{At}x_i + \int_0^t e^{A(t-\tau)}Bu_i(\tau)d\tau \in D \tag{2}$$

for all $t \in [0, \infty)$. Let

$$x(t) = \sum_{i=1}^{n+1} \lambda_i \left(e^{At}x_i + \int_0^t e^{A(t-\tau)}Bu_i(\tau)d\tau \right).$$

Clearly, $x(t) \in co(D)$. Define $u(t) = \sum_{i=1}^{n+1} \lambda_i u_i(t)$ for all $t \in [0, \infty)$. $u(\cdot)$ is measurable and $u(t) \in U$ for all $t \geq 0$, since U is convex. Moreover,

$$\begin{aligned} x(t) =& e^{At} \sum_{i=1}^{n+1} \lambda_i x_i + \int_0^t e^{A(t-\tau)} B \sum_{i=1}^{n+1} \lambda_i u_i(\tau) d\tau \\ =& e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau. \end{aligned}$$

Therefore, $x(\cdot)$ is a solution of the differential inclusion (1) starting at x_0 such that $x(t) \in co(D)$ for all $t \geq 0$. Hence, the set $co(D)$ is viable.

By Theorem 2, $\text{Viab}_F(K)$ is closed and viable and therefore so is $co(\text{Viab}_F(K))$. Since $\text{Viab}_F(K) \subseteq K$ and K is convex, then $co(\text{Viab}_F(K)) \subseteq K$. Therefore, by the maximality of the $\text{Viab}_F(K)$ (Theorem 2), $co(\text{Viab}_F(K)) \subseteq \text{Viab}_F(K)$. Since anyway $\text{Viab}_F(K) \subseteq co(\text{Viab}_F(K))$, $\text{Viab}_F(K)$ is convex. ∎

3 Viable polytopes

We now restrict our attention further to the following class of problems:

- Dynamics
$$\dot{x} = Ax + Bu \tag{3}$$
$x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$.

- Input constraints given by a convex, compact polytope
$$U = co\{\hat{u}_1, \ldots, \hat{u}_M\} \subseteq \mathbb{R}^m$$
$\hat{u}_i \in \mathbb{R}^m$, $i = 1, \ldots, M$.

- State constraints given by a convex, compact polytope
$$K = co\{\hat{k}_1, \hat{k}_2, \ldots, \hat{k}_N\} \subset \mathbb{R}^n$$
$\hat{k}_i \in \mathbb{R}^n$, $i = 1, \ldots, N$.

Notice that, even in this very restricted set up, the set $\text{Viab}_F(K)$ is not necessarily a polytope. We will try to compute viable polytopes contained in $\text{Viab}_F(K)$.

3.1 Basic facts

For a polytope $W = co\{w_1, \ldots, w_l\}$, $T_W(x)$ is easy to compute. A standard fact from set-valued analysis (see, for example, [13]) shows that for all $x \in W$, $y \in T_W(x)$ if and only if there exists $t > 0$ and $\lambda_i \geq 0$ with $\sum_{i=1}^l \lambda_i = 1$ such that $x + ty = \sum_{i=1}^l \lambda_i w_i$. In fact, we can choose any t satisfying $0 < t \leq \frac{1}{|y|} \min\{|x - w_i| \mid i = 1, \ldots, l\}$.

Lemma 1 *A polytope $W = co\{w_1, \ldots, w_l\}$ is viable for the system (3) if and only if $T_W(w_i) \cap F(w_i) \neq \emptyset$ for all $i = 1, \ldots, l$.*

Proof: For the second part, necessity is obvious from Theorem 1. For sufficiency, take any point $x \in W$ and some $y \in F(x)$. By definition, there exist $\mu_j \geq 0$, $j = 1, \ldots, M$ with $\sum_{j=1}^M \mu_j = 1$ such that

$$y = Ax + \sum_{j=1}^M \mu_j B\hat{u}_j.$$

Recall that $y \in T_W(x)$ if and only if there exist $t > 0$ and $\nu_j \geq 0$, $j = 1, \ldots, l$ with $\sum_{j=1}^l \nu_j = 1$ such that

$$y = \frac{1}{t}\left(\sum_{j=1}^l \nu_j w_j - x\right).$$

Therefore, $F(x) \cap T_W(x) \neq \emptyset$ if (and only if)

$$(A + \frac{1}{t}I)x = \frac{1}{t}\sum_{j=1}^l \nu_j w_j - \sum_{j=1}^M \mu_j B\hat{u}_j,$$

for some $t > 0$, $\nu_j \geq 0$, $j = 1, \ldots, l$ with $\sum_{j=1}^l \nu_j = 1$ and $\mu_j \geq 0$, $j = 1, \ldots, M$ with $\sum_{j=1}^M \mu_j = 1$.

Assume this holds for $x = w_i$, $i = 1, \ldots, l$, i.e.

$$(A + \frac{1}{t_i}I)w_i = \frac{1}{t_i}\sum_{j=1}^l \nu_{ij} w_j - \sum_{j=1}^M \mu_{ij} B\hat{u}_j.$$

Since W is polytope, $x + \bar{t}y \in W$ implies $x + ty \in W, \forall t \leq \bar{t}$. Let $t = \min\{t_1, \ldots, t_l\}$. Then

$$(A + \frac{1}{t}I)w_i = \frac{1}{t}\sum_{j=1}^l \nu_{ij} w_j - \sum_{j=1}^M \mu_{ij} B\hat{u}_j.$$

Consider an arbitrary $x \in W$. There exist $\lambda_i \geq 0$, $i = 1, \ldots, l$ with $\sum_{i=1}^l \lambda_i = 1$ such that

$$x = \sum_{i=1}^l \lambda_i w_i.$$

Taking a weighted average of the equations for the vertices,

$$\begin{aligned}(A + \frac{1}{t}I)x &= \frac{1}{t}\sum_{i=1}^l \lambda_i \sum_{j=1}^l \nu_{ij} w_j - \sum_{i=1}^l \lambda_i \sum_{j=1}^M \mu_{ij} B\hat{u}_j \\ &= \frac{1}{t}\sum_{j=1}^l \nu_j w_j - \sum_{j=1}^M \mu_j B\hat{u}_j\end{aligned}$$

with $\nu_j = \sum_{i=1}^l \lambda_i \nu_{ij}$ and $\mu_j = \sum_{i=1}^l \lambda_i \mu_{ij}$. Noting that $\nu_j \geq 0, j = 1, \ldots l$, $\mu_j \geq 0, j = 1, \ldots, M$ and

$$\sum_{j=1}^M \mu_j = \sum_{j=1}^M \sum_{i=1}^l \lambda_i \mu_{ij} = \sum_{i=1}^l \lambda_i \sum_{j=1}^M \mu_{ij} = 1$$

(and similarly for ν_j) proves the claim. ∎

From the proof of Lemma 1, we know that the polytope W is viable if and only if for each $i \in \{1, \ldots l\}$ there exists $t_i > 0$ small enough such that the following system of linear inequalities is consistent:

$$\begin{cases} (A + \frac{1}{t_i}I)w_i = \frac{1}{t_i}\sum_{j=1}^l \nu_{ij} w_j - \sum_{j=1}^M \mu_{ij} B\hat{u}_j \\ \sum_{j=1}^l \nu_{ij} = 1, \; \sum_{j=1}^M \mu_{ij} = 1 \\ \nu_{ij} \geq 0, j = 1, \ldots, l, \mu_{ij} \geq 0, j = 1, \ldots, M \end{cases} \tag{4}$$

Algorithm 1 (Viable Polytope Approximation)

```
W_0 = ∅
solve the linear equation
    Ax + Bu = 0
    subject to x ∈ K, u ∈ U
if a solution (x̂, û) ∈ K × U exists then
    i = 1
    W_i = {x̂}
    D = W_i
    while D ≠ ∅
        select a ∈ D
        W = W_i \ {a}
        solve the optimization problem
            min_{a_1,...,a_N} ∑_{j=1}^N |a_j − k̂_j|
            s.t.  a_j ∈ co{a, k̂_j}
                  T_{co(W ∪ {a_1,...,a_N})}(a_j) ∩ F(a_j)
                  ≠ ∅, j = 1, ..., N
        if co(W ∪ {a_1, ..., a_N}) ≠ co(W_i) then
            W_{i+1} = W ∪ {a_1, ..., a_N}
            D = W_{i+1}
            i = i + 1
        else
            D = D \ {a}
        endif
    endwhile
endif
```

Table 1: Algorithm

Determining the consistency of a system of linear inequalities can be transformed into solving an auxiliary linear programming problem. The linear inequalities $Nx \leq b$ are consistent if and only if the minimum value of the objective function of the linear programming problem:

$$\begin{aligned} &\min z \\ &\text{s.t. } Nx + (z, \ldots, z)^T \leq b \\ &\quad z \geq 0 \end{aligned}$$

is zero.

3.2 Computation of viable polytopes using optimization

Based on Lemma 1 we can give a method of computing a viable polytope for the system (3). Suppose that we have a viable polytope $W_i \subset K$. We then find r points $a_1, \ldots, a_r$ such that the polytope $W_{i+1} = co(W_i \bigcup \{a_1, \ldots, a_p\}) \subset K$ is still viable. In Algorithm 1, we choose $r = N$ and try to make a_i near the vertices, $\hat{k}_j$, of the set K. This is realized by minimizing $\sum_{j=1}^N |a_j - \hat{k}_j|$. It should be stressed that the structure of the algorithm is motivated by the need to generate proofs of its properties and not computational efficiency. Possible improvements include minimizing W_i every time new vertices are added to ensure there are no redundant vertices.

The algorithm starts by computing an *equilibrium point* of the system, in the sense of a state $x \in K$ for which there exists a $u \in U$ such that $Ax + Bu = 0$. The set of all equilibrium points (the *equilibrium set*) can be computed using the solution set of the linear inequalities:

$$\begin{cases} \sum_{j=1}^N \mu_j A\hat{k}_j + \sum_{i=1}^M \lambda_i B\hat{u}_i = 0 \\ \sum_{i=1}^M \lambda_i = 1, \lambda_i \geq 0, i = 1, \ldots, M \\ \sum_{j=1}^N \mu_j = 1, \mu_j \geq 0, j = 1, \ldots, N \end{cases} \tag{5}$$

by setting $x = \sum_{j=1}^N \mu_j \hat{k}_j$. The unknowns in the linear inequalities are μ_i, $i = 1, \ldots, M$ and λ_j, $j = 1, \ldots, N$ ($N + M$ in total). It is well known that the solution set of the above linear inequalities is a polytope. One could start the algorithm by computing the entire solution set and using it as the initial value of W_1. Alternatively, a single equilibrium point, $\hat{x}$ can be found by solving the following linear program:

$$\begin{aligned} &\min z \\ &\text{s.t. } \sum_{j=1}^N \mu_j A\hat{k}_j + \sum_{i=1}^M \lambda_i Bu_i = 0 \\ &\quad \sum_{i=1}^M \lambda_i = 1, \sum_{j=1}^N \mu_j = 1 \\ &\quad \lambda_i - z \geq 0, i = 1, \ldots, M, \mu_j - z \geq 0, j = 1, \ldots, N, z \geq 0 \end{aligned}$$

and setting $\hat{x} = \sum_{j=1}^N \mu_j \hat{k}_j$ if the optimal value $\hat{z}$ is zero, otherwise the solution set of the system (5) is empty. This is because that a solution of the above linear program must be a solution of the system (5).

The optimization problem needed for the remaining steps of the algorithm can be formulated so that it has a linear objective function and quadratic constraints. Assume that at the k^{th} step of the algorithm the set W_k is given by

$$W_k = \{w_1, \ldots, w_l, a\}$$

and let $W = W_k \backslash \{a\}$. Some algebraic manipulation shows that there exists $t > 0$ such that the optimization problem

$$\begin{aligned} &\min_{a_1, \ldots, a_N} \sum_{j=1}^N |a_j - \hat{k}_j| \\ &\text{s.t. } a_j \in co(\{a, \hat{k}_j\}), j = 1, \ldots, N \\ &\quad T_{co(W \cup \{a_1, \ldots, a_N\})}(a_j) \bigcap F(a_j) \neq \emptyset, j = 1, \ldots, N \end{aligned}$$

is equivalent to

$$\begin{aligned}
&\min_{p_1,\ldots,p_N} \sum_{j=1}^{N} p_j |a-\hat{k}_j| \\
&\text{s.t.}\ \frac{1}{t}\left(\sum_{i=1}^{l} s_i w_i + \sum_{i=l+1}^{N+l} s_i \hat{k}_{i-l} - \sum_{i=l+1}^{N+l} s_i p_{i-l}(\hat{k}_{i-l}-a)\right) \\
&\quad = \left(A-\frac{1}{t}I\right)(\hat{k}_j - p_j(\hat{k}_j - a)) + \sum_{i=1}^{M} \lambda_i B u_i \\
&\quad j=1,\ldots,N, 0 \le p_j \le 1, j=1,\ldots,N \\
&\quad \sum_{i=1}^{M} \lambda_i = 1, \lambda_i \ge 0, i=1,\ldots,M \\
&\quad \sum_{i=1}^{N+l} s_i = 1, s_i \ge 0, i=1,\ldots,N+l
\end{aligned}$$

with $a_j = p_j a + (1-p_j)\hat{k}_j$, $j=1,\ldots,N$. The variables in the latter optimization problem are s_i, $i=1,\ldots,N+l$, p_j, $j=1,\ldots,N$ and λ_i, $i=1,\ldots,M$ ($2N+M+l$ in total, where l depends on the step of the algorithm). The positive number t needs to be small enough. The problem has a linear objective, but the constraints are quadratic because of the products $s_i p_{i-l}$. Existing algorithms, for instance interior point methods [16], can be applied for solving this optimization problem.
An alternative version of the algorithm uses the faces (as opposed to vertices) of the polytope K. Assume that the polytope K is given in terms of the linear inequalities

$$K = \{x \in \mathbb{R}^n \mid Nx \le b\},$$

where N is an $m \times n$ matrix, $b \in \mathbb{R}^n$. Denote $N = (N_1^T, \ldots, N_m^T)^T$ and $b = (b_1, \ldots, b_m)^T$. In this case, we can replace the optimization problem in Algorithm 1 by the following:

$$\begin{aligned}
&\min_{a_1,\ldots,a_m} \sum_{j=1}^{m} |N_j a_j - b_j| \\
&\text{s.t.}\ N a_j \le b, j=1,\ldots,m \\
&\quad T_{co(W\cup\{a_1,\ldots,a_N\})}(a_j) \cap F(a_j) \ne \emptyset, j=1,\ldots,N
\end{aligned}$$

3.3 Analysis of the algorithm

First a trivial fact:

Lemma 2 *Consider $a \in co\{\hat{k}_1,\ldots,\hat{k}_N\}$ and take any $a_i \in co\{a,\hat{k}_i\}$, $i=1,\ldots,N$. Then $a \in co\{a_1,\ldots,a_N\}$.*

Proof: By construction $a = \sum_{i=1}^{N} \lambda_i \hat{k}_i$ and $a_i = \mu_i a + (1-\mu_i)\hat{k}_i$. If there exists $\mu_i = 1$ then $a_i = a$ and the claim is true. Otherwise

$$\hat{k}_i = \frac{1}{1-\mu_i}(a_i - \mu_i a).$$

Substituting into the expression for a and rearranging leads to

$$a = \sum_{i=1}^{N} \frac{\lambda_i}{(1-\mu_i)\sum_{j=1}^{N}\frac{\lambda_j}{1-\mu_j}} a_i.$$

It is easy to verify that the coefficients are non-negative and add up to 1. ∎

Next some basic properties of the algorithm:

Proposition 2 *If the algorithm terminates with $W = \emptyset$ then $Viab_F(K) = \emptyset$. Otherwise, for all i for which W_i is defined, $co(W_i)$ is viable and the optimization problem is feasible. Moreover, $co(W_i) \subseteq co(W_{i+1})$ whenever W_{i+1} is well-defined (i.e., unless the algorithm terminates at the i^{th} step).*

Proof: Termination with $W = \emptyset$ implies that $Ax+Bu=0$ has no solution in $K \times U$. By Theorem 3, K can not contain any non-empty, convex viable set. Since $\mathrm{Viab}_F(K)$ is viable and convex, it must be the empty set.
$co(W_0)$ is viable vacuously. If $Ax+Bu=0$ has a solution $(\hat{x},\hat{u}) \in K \times U$ then $co(W_1) = \{\hat{x}\}$ is also viable, since $T_{co(W_1)}(\hat{x}) = \{0\} = \{A\hat{x}+B\hat{u}\} \subseteq F(\hat{x})$.
Assume that $W_i = \{w_1,\ldots,w_l,a\}$ is viable for some i. Let $W = W_i \setminus \{a\}$ and consider the optimization problem

$$\begin{aligned}
&\min_{a_1,\ldots,a_N} \sum_{j=1}^{N} |a_j - \hat{k}_j| \\
&\text{s.t.}\ a_j \in co\{a,\hat{k}_j\} \\
&\quad T_{co(W\cup\{a_1,\ldots,a_N\})}(a_j) \cap F(a_j) \ne \emptyset \\
&\quad j=1,\ldots,N
\end{aligned}$$

The optimization problem is feasible. For example, let $a_j = a$ for all $j=1,\ldots,N$. Clearly $a_j \in co\{a,\hat{k}_j\}$. Moreover, $T_{W\cup\{a\}}(a) \cap F(a) \ne \emptyset$ since W_i is viable.
Let $\{a_1,\ldots,a_N\}$ be the optimal solution and assume that $co(W\cup\{a_1,\ldots,a_N\}) \ne co(W_i)$. Then

$$\begin{aligned}
W_{i+1} =& W \cup \{a_1,\ldots,a_N\} \\
=& \{w_1,\ldots,w_l,a_1,\ldots,a_N\}.
\end{aligned}$$

We first show that $co(W_i) \subseteq co(W_{i+1})$. Take any $x \in co(W_i)$. There exist $\lambda_i \ge 0$, $i=0,\ldots,l$, $\sum_{i=0}^{l}\lambda_i = 1$ such that

$$x = \lambda_0 a + \sum_{i=1}^{N} \lambda_i w_i.$$

By Lemma 2, $a \in co\{a_1,\ldots,a_N\}$. Therefore, there exists $\mu_i \ge 0$, $i=1,\ldots,N$, $\sum_{i=1}^{N}\mu_i = 1$ such that

$$x = \lambda_0 \sum_{j=1}^{N} \mu_i a_i + \sum_{i=1}^{l} \lambda_i w_i.$$

Clearly $\lambda_i, \lambda_0, c_j \ge 0$ and add up to 1. Therefore,

$$x \in co\{w_1,\ldots,w_l,a_1,\ldots,a_N\} = co(W_{i+1}).$$

Finally, we show that $co(W_{i+1})$ is viable. Since $co(W_i)$ is viable and $co(W_i) \subseteq co(W_{i+1})$,

$$T_{co(W_{i+1})}(w_i) \cap F(w_i) \neq \emptyset.$$

By construction,

$$T_{co(W_{i+1})}(a_j) \cap F(a_j) \neq \emptyset.$$

Therefore, W_{i+1} is viable by Lemma 1. The claim follows by induction. ■

4 Concluding remarks

We presented an algorithm for computing viable polytopic subsets of a given polytope under linear dynamics. The main feature of the proposed algorithm is that at each step it maintains as its state a polytopic subset of the given set that is viable. Therefore, if the algorithm is terminated before completion, its state can still be used for safe controller design.

Clearly the algorithm needs to be extended in many directions. One extension is to unbounded polyhedra; work in this direction is currently underway. Even for bounded polytopes it is easy to see that the algorithm often fails to produce the maximal viable polytope. For example, the algorithm sometimes gives conservative results when applied to systems where the maximal viable polytope has empty interior. When applied to the system

$$A = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix},\ B = \begin{bmatrix} 0 \\ 0 \end{bmatrix},\ K = [-1,1] \times [-1,1].$$

our algorithm terminates at one step with $W_1 = \{0\}$. The maximal viable polytope in this case is $W = [-1,1] \times \{0\}$ is viable and has empty interior. The algorithm may also be conservative when applied to systems that admit ellipsoidal viable sets but not polyhedral viable sets (or only trivial ones). For example when applied to the system

$$A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},\ B = \begin{bmatrix} 0 \\ 0 \end{bmatrix},\ K = [-1,1] \times [-1,1].$$

The algorithm terminates at one step with $W_1 = \{0\}$. In this case the viability kernel is the unit disc, which does not contain any viable polytopes (other than the trivial $\{0\}$). To deal with systems like these we would like to extend our approach to include ellipsoidal constraints.

References

[1] J.-P. Aubin. "Viability Theory", Birkhäuser, Boston, (1991).

[2] J.-P. Aubin. "Optima and Equilibria: An Introduction to Nonlinear Analysis", Springer-Verlag, Berlin, (1993).

[3] J.-P. Aubin and H. Frankowska. "Set Valued Analysis", Birkhäuser, Boston, (1990).

[4] G. Bitsoris. "Existence of Positively invariant polyhedral sets for continuous-time linear systems", Control-Theory and Advanced Technology, **7**, pp. 407-427, (1991).

[5] F. Blanchini. "Set invariance in control", Automatica, **35**, pp. 1747-1767, (1999).

[6] A. Bemporad, L. Giovanardi, and F. D. Torrisi. "Performance driven reachability analysis for optimal scheduling and control of hybrid systems", in Proc. 39th IEEE Conference on Decision and Control, pp. 965-974, (2000).

[7] L. Berardi, E. De Santis, and M. D. Di Benedetto. "A structural approach to the control of switching systems with an application to automotive engines", in Proc. 38th IEEE Conference on Decision and Control, pp. 3526–3531, (1999).

[8] E.B. Castelan and J.C. Hennet. "On invariant polyhedra of continuous-time linear system", IEEE Trans. Automatic Control, **38**, pp. 1680-1685, (1993).

[9] L.C.G.J.M. Habets and J.H. van Schuppen. "A controllability result for piecewise linear hybrid systems", in Proc. 6th European Control Conference, pp. 3870–3873, (2001).

[10] X.D. Koutsoukos and P.J. Antsaklis. "Design of hybrid system regulators", in Proc. 37th IEEE Conference on Decision and Control, pp. 3990-3995, (1999).

[11] B.E.A. Milani and C.E.T. Dorea. "On invariant polyhedra of continuous-time linear systems subject to additive disturbances", Automatica, **32**, pp. 785-789, (1996).

[12] R.T. Rockafellar and R. J.-B. Wets. "Variational analysis", Springer-Verlag, New York, (1998).

[13] R.T. Rockafellar. "Convex Analysis", Princeton University Press, Princeton, (1970).

[14] S. Tarbouriech and C. Burgat. "Positively invariant sets for constrained continuous-time systems with cone properties", IEEE Trans. Automatica Control, **39**, pp. 401-405, (1994).

[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 Conference on Decision and Control, pp. 1243-1248, (2001).

[16] S. J. Wright. "Primal-Dual Interior-Point Methods", SIAM, Philadephia, (1997).