Reachability Analysis of Hybrid Systems Using Support Functions*

Colas Le Guernic[1] and Antoine Girard[2]

[1] Verimag, Université de Grenoble
[2] Laboratoire Jean Kuntzmann, Université de Grenoble
{Colas.Le-Guernic,Antoine.Girard}@imag.fr

Abstract. This paper deals with conservative reachability analysis of a class of hybrid systems with continuous dynamics described by linear differential inclusions, convex invariants and guards, and linear reset maps. We present an approach for computing over-approximations of the set of reachable states. It is based on the notion of support function and thus it allows us to consider invariants, guards and constraints on continuous inputs and initial states defined by arbitrary compact convex sets. We show how the properties of support functions make it possible to derive an effective algorithm for approximate reachability analysis of hybrid systems. We use our approach on some examples including the navigation benchmark for hybrid systems verification.

1 Introduction

Reachability analysis has been a major research issue in the field of hybrid systems over the past decade. An important part of the effort has been devoted to hybrid systems where the continuous dynamics is described by linear differential equations or inclusions. This work resulted in several computational techniques for approximating the reachable set of a hybrid system using several classes of convex sets including polyhedrons [1,2], ellipsoids [3,4], hyperrectangles [5] or zonotopes [6,7]. In these approaches, the set of continuous inputs is assumed to belong to the considered class; invariants and guards are usually given by polyhedrons or also sometimes by ellipsoids (e.g. in [3]).

In this paper, we propose an approach that can handle hybrid systems where invariants, guards and constraints on the continuous inputs and initial states are defined by arbitrary compact convex sets. It is based on the representation of compact convex sets using their support function. Algorithms based on support functions have already been proposed for reachability analysis of purely continuous systems in [8] and more recently in [9], using the efficient computational scheme presented in [7]. We extend this latter approach to handle hybrid dynamics. The paper is organized as follows. In section 2, we briefly present some results from [9] on reachability analysis of continuous linear systems. In section 3, we adapt these results for hybrid systems by taking care of the constraints

* This work was supported by the ANR SETIN project VAL-AMS.

A. Bouajjani and O. Maler (Eds.): CAV 2009, LNCS 5643, pp. 540–554, 2009.

imposed by invariants and guards. In section 4, we present the notion of support functions for convex sets. In section 5, we show how the properties of support functions make it possible to derive an effective implementation of the algorithm presented in section 3 for reachability analysis of hybrid systems. Finally, we use our approach on some examples including the navigation benchmark for hybrid systems verification [10].

Notations : Given a set $\mathcal{S} \subseteq \mathbb{R}^n$, $\mathrm{CH}(\mathcal{S})$ denotes its convex hull. For a matrix M, $M\mathcal{S}$ denotes the image of $\mathcal{S}$ by M, and for a real number λ, $\lambda\mathcal{S} = (\lambda I)\mathcal{S}$ where I is the identity matrix. For $\mathcal{S}$, $\mathcal{S}' \subseteq \mathbb{R}^n$, $\mathcal{S} \oplus \mathcal{S}'$ denotes the Minkowski sum of $\mathcal{S}$ and $\mathcal{S}'$: $\mathcal{S} \oplus \mathcal{S}' = \{x + x' : \ x \in \mathcal{S}, x' \in \mathcal{S}'\}$. For a matrix M, $M^\top$ denotes its transpose.

2 Reachability Analysis of Linear Systems

In this paper, we shall consider a class of hybrid systems where the continuous dynamics is described by linear differential inclusions of the form:

$$\dot{x}(t) \in Ax(t) \oplus \mathcal{U},$$

where the continuous state $x(t) \in \mathbb{R}^n$, A is a $n \times n$ matrix and $\mathcal{U} \subseteq \mathbb{R}^n$ is a compact convex set; note that $\mathcal{U}$ need not be full dimensional. Let $\mathcal{X} \subseteq \mathbb{R}^n$, we denote by $\mathcal{R}_{\mathrm{C}}(s, \mathcal{X}) \subseteq \mathbb{R}^n$ the set of states reachable at time $s \geq 0$ from states in $\mathcal{X}$: $\mathcal{R}_{\mathrm{C}}(s, \mathcal{X}) = \{x(s) : \ \forall t \in [0, s], \ \dot{x}(t) \in Ax(t) \oplus \mathcal{U}, \text{ and } x(0) \in \mathcal{X}\}$. Then, the reachable set on the time interval $[s, s']$ is defined as

$$\mathcal{R}_{\mathrm{C}}([s, s'], \mathcal{X}) = \bigcup_{t \in [s, s']} \mathcal{R}_{\mathrm{C}}(t, \mathcal{X}).$$

Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a compact convex set of initial states and $T > 0$ a time bound. In the recent paper [9], we presented an improved computational scheme, adapted from [6], for the over-approximation of the reachable set $\mathcal{R}_{\mathrm{C}}([0, T], \mathcal{X})$. Let $\|.\|$ be a norm and $\mathcal{B} \subseteq \mathbb{R}^n$ the associated unit ball. We denote $\delta_{\mathcal{X}} = \max_{x \in \mathcal{X}} \|x\|$, and $\delta_{\mathcal{U}} = \max_{u \in \mathcal{U}} \|u\|$. We use a discretization of the time with step $\tau = T/N$ ($N \in \mathbb{N}$). We have

$$\mathcal{R}_{\mathrm{C}}([0, T], \mathcal{X}) = \bigcup_{i=0}^{N-1} \mathcal{R}_{\mathrm{C}}([i\tau, (i+1)\tau], \mathcal{X}).$$

An over-approximation of $\mathcal{R}_{\mathrm{C}}([0, T], \mathcal{X})$ can be obtained by computing over-approximations of the sets $\mathcal{R}_{\mathrm{C}}([i\tau, (i+1)\tau], \mathcal{X})$. We consider the first element of the sequence.

Lemma 1. *[9] Let $\mathcal{Y}_0 \subseteq \mathbb{R}^n$ be defined by :*

$$\mathcal{Y}_0 = \mathrm{CH}\left(\mathcal{X} \cup (e^{\tau A}\mathcal{X} \oplus \tau\mathcal{U} \oplus \alpha_\tau \mathcal{B})\right) \tag{1}$$

where $\alpha_\tau = (e^{\tau\|A\|} - 1 - \tau\|A\|)(\delta_{\mathcal{X}} + \frac{\delta_{\mathcal{U}}}{\|A\|})$. Then, $\mathcal{R}_{\mathrm{C}}([0, \tau], \mathcal{X}) \subseteq \mathcal{Y}_0$.

This lemma can be roughly understood as follows, $e^{\tau A}\mathcal{X}\oplus\tau\mathcal{U}$ is an approximation of the reachable set at time τ; a bloating operation followed by a convex hull operation gives an approximation of $\mathcal{R}_{\mathrm{C}}([0,\tau],\mathcal{X})$. The bloating factor α_τ is chosen to ensure over-approximation. We consider the other elements of the sequence. Let us remark that

$$\mathcal{R}_{\mathrm{C}}([(i+1)\tau,(i+2)\tau],\mathcal{X}) = \mathcal{R}_{\mathrm{C}}\left(\tau,\mathcal{R}_{\mathrm{C}}(,[i\tau,(i+1)\tau],\mathcal{X})\right),\ i=0,\dots,N-2.$$

For $\mathcal{Y}\subseteq\mathbb{R}^n$, the following lemma gives an over-approximation of $\mathcal{R}_{\mathrm{C}}(\tau,\mathcal{Y})$:

Lemma 2. *[9] Let $\mathcal{Y}\subseteq\mathbb{R}^n$, $\mathcal{Y}'\subseteq\mathbb{R}^n$ defined by $\mathcal{Y}'=e^{\tau A}\mathcal{Y}\oplus\tau\mathcal{U}\oplus\beta_\tau\mathcal{B}$ where $\beta_\tau=(e^{\tau\|A\|}-1-\tau\|A\|)\frac{\delta_\mathcal{U}}{\|A\|}$. Then, $\mathcal{R}_{\mathrm{C}}(\tau,\mathcal{Y})\subseteq\mathcal{Y}'$.*

The set $e^{\tau A}\mathcal{Y}\oplus\tau\mathcal{U}$ is an approximation the reachable set at time τ; bloating this set using the ball of radius β_τ ensures over-approximation. We can define the compact convex sets $\mathcal{Y}_i$ over-approximating the reachable sets $\mathcal{R}_{\mathrm{C}}([i\tau,(i+1)\tau],\mathcal{X})$ as follows. $\mathcal{Y}_0$ is given by equation (1) and

$$\mathcal{Y}_{i+1}=e^{\tau A}\mathcal{Y}_i\oplus\tau\mathcal{U}\oplus\beta_\tau\mathcal{B},\ i=0,\dots,N-2. \tag{2}$$

Then, it follows from Lemmas 1 and 2:

Proposition 1. *[9] Let us consider the sets $\mathcal{Y}_0,\dots,\mathcal{Y}_{N-1}$ defined by equations (1) and (2); then, $\mathcal{R}_{\mathrm{C}}([0,T],\mathcal{X})\subseteq(\mathcal{Y}_0\cup\dots\cup\mathcal{Y}_{N-1})$.*

We refer the reader to our work in [9] for technical proofs and explicit bounds on the approximation error.

3 Reachability Analysis of Hybrid Systems

In this paper, we consider a class of hybrid systems with continuous dynamics described by linear differential inclusions, convex invariants and guards, and linear reset maps. Formally, a hybrid system is a tuple $\mathrm{H}=(\mathbb{R}^n,Q,E,F,\mathcal{I},\mathcal{G},R)$ where $\mathbb{R}^n$ is the continuous state-space, Q is a finite set of locations and $E\subseteq Q\times Q$ is the set of discrete transitions. $F=\{F_q:q\in Q\}$ is a collection of continuous dynamics; for each $q\in Q$, $F_q=(A_q,\mathcal{U}_q)$ where A_q is a $n\times n$ matrix and $\mathcal{U}_q\subseteq\mathbb{R}^n$ is a compact convex set. $\mathcal{I}=\{\mathcal{I}_q:q\in Q\}$ is a collection of invariants; for each $q\in Q$, $\mathcal{I}_q\subseteq\mathbb{R}^n$ is a compact convex set. $\mathcal{G}=\{\mathcal{G}_e:e\in E\}$ is a collection of guards; for each $e\in E$, $\mathcal{G}_e\subseteq\mathbb{R}^n$ is either a compact convex set or a hyperplane. $R=\{R_e:e\in E\}$ is a collection of reset maps; for each $e\in E$, $R_e=(B_e,\mathcal{V}_e)$ where B_e is a $n\times n$ matrix and $\mathcal{V}_e\subseteq\mathbb{R}^n$ is a compact convex set. Let us remark that the sets $\mathcal{U}_q$, $\mathcal{I}_q$, $\mathcal{G}_e$ and $\mathcal{V}_e$ are only assumed to be compact and convex, these can be polyhedrons, ellipsoids or more complex convex sets. We distinguish the case where $\mathcal{G}_e$ is a hyperplane; in this case, we shall see that reachability analysis can be processed more accurately.

The state of the hybrid system at time t is a pair $(q(t),x(t))$ consisting of a discrete state $q(t)\in Q$ and a continuous state $x(t)\in\mathbb{R}^n$. The state of the hybrid system can evolve in a continuous or discrete manner. During the continuous evolution, the discrete state remains constant $q(t) = q$ and the continuous state evolves according to the linear differential inclusion:

$$\dot{x}(t) \in A_q x(t) \oplus \mathcal{U}_q \text{ and } x(t) \in \mathcal{I}_q.$$

The discrete evolution is enabled at time t if the state $(q(t), x(t)) = (q, x)$ satisfies $x \in \mathcal{G}_e$ for some $e = (q, q') \in E$. Then, the transition e can occur instantaneously: the state of the hybrid system jumps to $(q(t), x(t)) = (q', x')$ where

$$x' \in B_e x \oplus \mathcal{V}_e.$$

In the following, for simplicity of the presentation, we consider reachability analysis of hybrid systems on an unbounded time interval with the understanding that the termination of the described algorithms is not ensured.

Let $\Sigma_0 = \bigcup_{q \in Q} \{q\} \times \mathcal{X}_{0,q} \subseteq Q \times \mathbb{R}^n$ be a set of initial states where, for $q \in Q$, $\mathcal{X}_{0,q} \subseteq \mathbb{R}^n$ is a compact convex set. The set of states reachable under the evolution of the hybrid system from this set of initial states is denoted $\mathcal{R}_{\mathrm{H}}(\Sigma_0)$. In a given location q, the set of continuous states reachable under the continuous evolution from a set of continuous states $\mathcal{X}$ is

$$\mathcal{R}_{\mathrm{loc}}(q, \mathcal{X}) = \{x(s) : s \geq 0,\ \forall t \in [0, s],\ \dot{x}(t) \in A_q x(t) \oplus \mathcal{U}_q,\ x(t) \in \mathcal{I}_q,\ x(0) \in \mathcal{X}\}$$

A transition $e \in E$ of the form $e = (q, q')$ can occur if $\mathcal{R}_{\mathrm{loc}}(q, \mathcal{X}) \cap \mathcal{G}_e \neq \emptyset$, the set of continuous states reachable just after the transition is

$$\mathcal{R}_{\mathrm{jump}}(e, \mathcal{X}) = B_e \left(\mathcal{R}_{\mathrm{loc}}(q, \mathcal{X}) \cap \mathcal{G}_e\right) \oplus \mathcal{V}_e.$$

Then, the set of reachable states $\mathcal{R}_{\mathrm{H}}(\Sigma_0)$ can be computed by Algorithm 1.

Algorithm 1. Reachability analysis of a hybrid system

Input: Set of initial states $\Sigma_0 = \bigcup_{q \in Q} \{q\} \times \mathcal{X}_{0,q}$
Output: $\mathcal{R} = \mathcal{R}_{\mathrm{H}}(\Sigma_0)$

$L \leftarrow \{(q, \mathcal{X}_{0,q}) : q \in Q\}$
$\Sigma \leftarrow \emptyset$
$\mathcal{R} \leftarrow \emptyset$
while $L \neq \emptyset$ **do**
  Pick $(q, \mathcal{X}) \in L$
  $\Sigma \leftarrow \Sigma \cup (\{q\} \times \mathcal{X})$
  $\mathcal{R} \leftarrow \mathcal{R} \cup (\{q\} \times \mathcal{R}_{\mathrm{loc}}(q, \mathcal{X}))$ ▷ Reachable set by continuous evolution
  for $e \in E$ of the form $e = (q, q')$ **do** ▷ Reachable set by discrete evolution
    if $\{q'\} \times \mathcal{R}_{\mathrm{jump}}(e, \mathcal{X}) \not\subseteq \Sigma$ **then**
      Insert $(q', \mathcal{R}_{\mathrm{jump}}(e, \mathcal{X}))$ in L ▷ Insert in L if not explored yet
    end if
  end for
end while
return $\mathcal{R}$

In the variable L, we store a list of sets from which reachability analysis has to be processed. $\Sigma \subseteq Q \times \mathbb{R}^n$ represents the set of explored states from which reachability analysis has already been made.

It is clear that an algorithm for computing over-approximations of $\mathcal{R}_{\text{loc}}(q, \mathcal{X})$ and $\mathcal{R}_{\text{jump}}(e, \mathcal{X})$ is sufficient for conservative reachability analysis of the hybrid system. This can be done using Algorithm 2 adapted from the method presented in the previous section for reachability analysis of linear systems. For lighter notations, the index q has been dropped. We use a discretization of time with a step $\tau > 0$. The real numbers α_τ and β_τ are those defined in Lemmas 1 and 2. Note that Algorithm 2 takes a convex set $\mathcal{X}$ as input; therefore, in order to use it in a straightforward implementation of Algorithm 1, it needs to compute a convex over-approximation of $\mathcal{R}_{\text{jump}}(e, \mathcal{X})$.

Algorithm 2. Reachability analysis in a given location q

Input: Convex set of states $\mathcal{X}$, time step τ.
Output: $\mathcal{R} \supseteq \mathcal{R}_{\text{loc}}(q, \mathcal{X})$; convex $\mathcal{X}_e \supseteq \mathcal{R}_{\text{jump}}(e, \mathcal{X})$, for $e \in E$ such that $e = (q, q')$

```
Z_0 ← CH (X ∪ (e^{τA} X ⊕ τU ⊕ α_τ B)) ∩ I          ▷ Initialize the reachable set
R ← Z_0
i ← 0
while Z_i ≠ ∅ do
    Z_{i+1} ← (e^{τA} Z_i ⊕ τU ⊕ β_τ B) ∩ I            ▷ Propagate the reachable set
    R ← R ∪ Z_{i+1}
    i ← i + 1
end while
for e ∈ E such that e = (q, q') do
    H_e ← CH(R ∩ G_e)                                  ▷ Intersect with guards
    X_e ← B_e H_e ⊕ V_e                                ▷ Reachable set after the transition
end for
return R; X_e, for e ∈ E such that e = (q, q')
```

Proposition 2. *Let $\mathcal{R}$; $\mathcal{X}_e$, for $e \in E$ such that $e = (q, q')$ be computed by Algorithm 2. Then, $\mathcal{R}_{\text{loc}}(q, \mathcal{X}) \subseteq \mathcal{R}$ and $\mathcal{R}_{\text{jump}}(e, \mathcal{X}) \subseteq \mathcal{X}_e$.*

Proof. Let $z \in R_{\text{loc}}(q, \mathcal{X})$, then there exists $s \geq 0$ and a function $x(.)$ such that $x(s) = z$, $x(0) \in \mathcal{X}$ and for all $t \in [0, s]$, $\dot{x}(t) \in Ax(t) \oplus \mathcal{U}$ and $x(t) \in \mathcal{I}$. Let $i^* \in \mathbb{N}$ such that $s \in [i^*\tau, (i^*+1)\tau]$. Let us remark that $s - i^*\tau \in [0, \tau]$, then from Lemma 1 and since $x(s - i^*\tau) \in \mathcal{I}$, it follows that $x(s - i^*\tau) \in \mathcal{Z}_0$. Let us show, by induction, that for all $i = 0, \dots, i^*$, $x(s + (i - i^*)\tau) \in \mathcal{Z}_i$. This is true for $i = 0$; let us assume that it is true for some $i \leq i^* - 1$. Then, from Lemma 2 and since $x(s + (i + 1 - i^*)\tau) \in \mathcal{I}$, it follows that $x(s + (i + 1 - i^*)\tau) \in \mathcal{Z}_{i+1}$. Therefore, for all $i = 0, \dots, i^*$, $x(s + (i - i^*)\tau) \in \mathcal{Z}_i$, which implies for $i = i^*$ that $z = x(s) \in \mathcal{Z}_{i^*} \subseteq \mathcal{R}$. The first part of the proposition is proved. It follows that $\mathcal{R}_{\text{jump}}(e, \mathcal{X}) \subseteq (B_e(\mathcal{R} \cap G_e) \oplus \mathcal{V}_e) \subseteq \mathcal{X}_e$. ■

In the following, we discuss the implementation of the Algorithm 2 based on the notion of support functions.

4 Support Functions of Convex Sets

The support function of a convex set is a classical tool of convex analysis. Support functions can be used as a representation of arbitrary complex compact convex sets. In this section, we present some properties of support functions and show how they can be used for the computation of polyhedral over-approximations of convex sets. The results are stated without the proofs that can be found in several textbooks on convex analysis (see e.g. [11,12,13]).

Definition 1. *Let $\mathcal{S} \subseteq \mathbb{R}^n$ be a compact convex set; the support function of $\mathcal{S}$ is $\rho_{\mathcal{S}} : \mathbb{R}^n \to \mathbb{R}$ defined by $\rho_{\mathcal{S}}(\ell) = \max_{x \in \mathcal{S}} \ \ell \cdot x$.*

The notion of support function is illustrated in Figure 1.

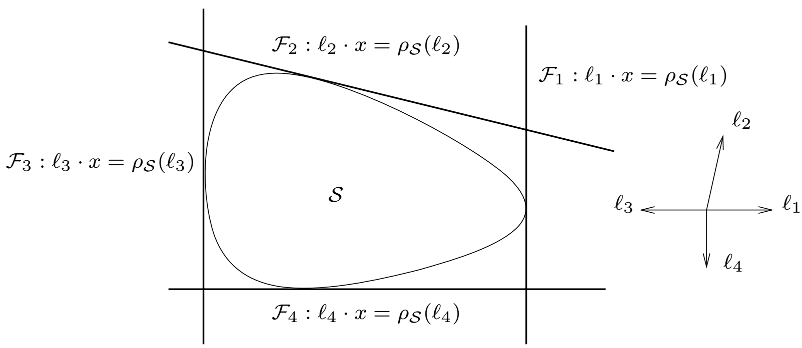


Fig. 1. Illustration of the notion of support function of a convex set $\mathcal{S}$

4.1 Properties of Support Functions

It can be shown that the support function of a compact convex set is a convex function. For two compact convex sets, $\mathcal{S}$ and $\mathcal{S}'$, it is easy to see that $\mathcal{S} \subseteq \mathcal{S}'$ if and only if $\rho_{\mathcal{S}}(\ell) \leq \rho_{\mathcal{S}'}(\ell)$ for all $\ell \in \mathbb{R}^n$. It is to be noted that a compact convex set $\mathcal{S}$ is uniquely determined by its support function as the following equality holds:

$$\mathcal{S} = \bigcap_{\ell \in \mathbb{R}^n} \{x \in \mathbb{R}^n : \ell \cdot x \leq \rho_{\mathcal{S}}(\ell)\}.$$

From the previous equation, it is clear that a polyhedral over-approximation of a compact convex set can be obtained by "sampling" its support function:

Proposition 3. *Let $\mathcal{S}$ be a compact convex set and $\ell_1, \ldots, \ell_r \in \mathbb{R}^n$ be arbitrarily chosen vectors; let us define the following polyhedron:*

$$\overline{\mathcal{S}} = \{x \in \mathbb{R}^n : \ell_k \cdot x \leq \rho_{\mathcal{S}}(\ell_k),\ k = 1, \ldots, r\}.$$

Then, $\mathcal{S} \subseteq \overline{\mathcal{S}}$. Moreover, we say that this over-approximation is tight as $\mathcal{S}$ touches the faces $\mathcal{F}_1, \ldots, \mathcal{F}_r$ of $\overline{\mathcal{S}}$.

An example of such polyhedral over-approximation of a convex set can be seen in Figure 1. The support function can be computed efficiently for a large class of compact convex sets. Let $\mathcal{B}_1$, $\mathcal{B}_2$, $\mathcal{B}_\infty$ denote the unit balls for the 1, 2 and ∞ norms. Then, $\rho_{\mathcal{B}_1}(\ell) = \|\ell\|_\infty$, $\rho_{\mathcal{B}_2}(\ell) = \|\ell\|_2$ and $\rho_{\mathcal{B}_\infty}(\ell) = \|\ell\|_1$. Let Q be a $n \times n$ positive definite symmetric matrix, then for the ellipsoid:

$$\mathcal{E} = \left\{x \in \mathbb{R}^n : x^\top Q^{-1} x \leq 1\right\},\ \rho_{\mathcal{E}}(\ell) = \sqrt{\ell^\top Q \ell}.$$

Let $g_1, \dots, g_r \in \mathbb{R}^n$, then for the zonotope:

$$\mathcal{Z} = \{\alpha_1 g_1 + \cdots + \alpha_r g_r : \alpha_j \in [-1,1],\ j = 1, \dots, r\},\ \rho_{\mathcal{Z}}(\ell) = \sum_{j=1}^{r} |g_j \cdot \ell|.$$

Let $c_1, \dots, c_r \in \mathbb{R}^n$ and $d_1, \dots, d_r \in \mathbb{R}$, then for the polyhedron

$$\mathcal{P} = \{x \in R^n :\ c_j \cdot x \leq d_j,\ j = 1, \dots, r\},$$

$\rho_{\mathcal{P}}(\ell)$ can be determined by solving the linear program

$$\begin{cases} \text{maximize } \ell \cdot x \\ \text{subject to } c_j \cdot x \leq d_j,\ j = 1, \dots, r \end{cases}$$

More complex sets can be considered using the following operations on elementary compact convex sets:

Proposition 4. *For all compact convex sets $\mathcal{S}$, $\mathcal{S}' \subseteq \mathbb{R}^n$, for all matrices M, all positive scalars λ, and all vectors ℓ of suitable dimension, the following assertions hold:*

$$\begin{aligned} \rho_{M\mathcal{S}}(\ell) &= \rho_{\mathcal{S}}(M^\top \ell) \\ \rho_{\lambda\mathcal{S}}(\ell) &= \rho_{\mathcal{S}}(\lambda\ell) = \lambda\rho_{\mathcal{S}}(\ell) \\ \rho_{\mathrm{CH}(\mathcal{S}\cup\mathcal{S}')}(\ell) &= \max(\rho_{\mathcal{S}}(\ell), \rho_{\mathcal{S}'}(\ell)) \\ \rho_{\mathcal{S}\oplus\mathcal{S}'}(\ell) &= \rho_{\mathcal{S}}(\ell) + \rho_{\mathcal{S}'}(\ell) \\ \rho_{\mathcal{S}\cap\mathcal{S}'}(\ell) &\leq \min\left(\rho_{\mathcal{S}}(\ell), \rho_{\mathcal{S}'}(\ell)\right) \end{aligned}$$

Except for the last property, these relations are all exact. For the intersection, we only have an over-approximation relation. The inequality comes from the fact that the function $\min\left(\rho_{\mathcal{S}}(\ell), \rho_{\mathcal{S}'}(\ell)\right)$ may not be a convex function. An exact relation between $\rho_{\mathcal{S}\cap\mathcal{S}'}$, $\rho_{\mathcal{S}}$ and $\rho_{\mathcal{S}'}$ exists[1]; unfortunately, this relation is not effective from the computational point of view. Let us remark, though, that for a convex set $\mathcal{K}$, such that $\rho_{\mathcal{K}}(\ell) \leq \min\left(\rho_{\mathcal{S}}(\ell), \rho_{\mathcal{S}'}(\ell)\right)$, for all $\ell \in \mathbb{R}^n$, it follows that $\mathcal{K} \subseteq \mathcal{S}$ and $\mathcal{K} \subseteq \mathcal{S}'$, thus $\mathcal{K} \subseteq \mathcal{S} \cap \mathcal{S}'$.

We shall see, further in the paper, how the properties presented in this section allow us to compute an over-approximation of the set $\mathcal{R}_{\mathrm{loc}}(q, \mathcal{X})$, given as the union of convex polyhedrons.

[1] Indeed, it can be shown [13] that $\rho_{\mathcal{S}\cap\mathcal{S}'}(\ell) = \inf_{w\in\mathbb{R}^n}\left(\rho_{\mathcal{S}}(\ell - w) + \rho_{\mathcal{S}'}(w)\right)$.

4.2 Intersection of a Compact Convex Set and a Hyperplane

We now consider the problem of computing the support function of the intersection of a compact convex set $\mathcal{S}$, given by its support function $\rho_{\mathcal{S}}$, and a hyperplane $\mathcal{G} = \{x \in \mathbb{R}^n : c \cdot x = d\}$ where $c \in \mathbb{R}^n$ and $d \in \mathbb{R}$. This will be useful to compute a polyhedral over-approximation of the intersection of $\mathcal{R}_{\text{loc}}(q, \mathcal{X})$ with a guard of the hybrid system given by a hyperplane. First of all, let us remark that checking whether $\mathcal{S} \cap \mathcal{G}$ is empty is an easy problem. Indeed, it is straightforward (see Figure 2) that $\mathcal{S} \cap \mathcal{G} \neq \emptyset$ if and only if $-\rho_{\mathcal{S}}(-c) \leq d \leq \rho_{\mathcal{S}}(c)$.

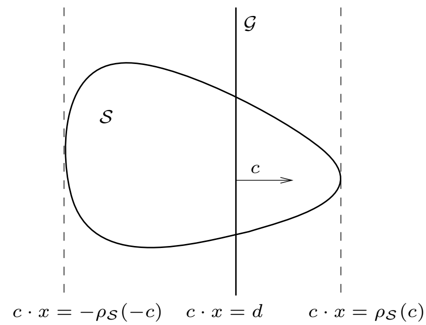


Fig. 2. Checking emptiness of $\mathcal{S} \cap \mathcal{G}$

In the following, we shall assume that $\mathcal{S} \cap \mathcal{G} \neq \emptyset$. Let $\ell \in \mathbb{R}^n$, we consider the problem of computing an accurate over-approximation of $\rho_{\mathcal{S}\cap\mathcal{G}}(\ell)$. The following result, adapted from [14], shows that the problem can be reduced to a two-dimensional problem by projecting on the subspace spanned by c and ℓ.

Proposition 5. *[14] Let $\Pi : \mathbb{R}^n \to \mathbb{R}^2$ be the projection operator defined by $\Pi x = (c \cdot x, \ell \cdot x)$. Then,* $\rho_{\mathcal{S}\cap\mathcal{G}}(\ell) = \max\{y_2 \in \mathbb{R} : (d, y_2) \in \Pi\mathcal{S}\}$.

Thus, the computation of $\rho_{\mathcal{S}\cap\mathcal{G}}(\ell)$ is reduced to a two dimensional optimization problem which essentially consists in computing the intersection of the two dimensional compact convex set $\Pi\mathcal{S}$ with the line $\mathcal{D} = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 = d\}$. We shall further reduce the problem. Let $\theta \in]0, \pi[$ and $v_\theta = (\cos\theta, \sin\theta)$, the equation of the line supporting $\Pi\mathcal{S}$ in the direction v_θ is $y_1 \cos\theta + y_2 \sin\theta = \rho_{\Pi\mathcal{S}}(v_\theta) = \rho_{\mathcal{S}}(\Pi^\top v_\theta)$. This line intersects the line $\mathcal{D}$ at the point of coordinates (y_1, y_2) with $y_1 = d$, $y_2 = (\rho_{\mathcal{S}}(\Pi^\top v_\theta) - d\cos\theta)/\sin\theta$, as shown on Figure 3. Then, let us define the function $f :]0, \pi[\to \mathbb{R}$ given by

$$f(\theta) = \frac{\rho_{\mathcal{S}}(\Pi^\top v_\theta) - d\cos\theta}{\sin\theta}.$$

It is easy to see that f is unimodal and that $\inf_{\theta\in]0,\pi[} f(\theta) = \sup\{y_2 \in \mathbb{R} : (d, y_2) \in \Pi\mathcal{S}\}$. Therefore, using a minimization algorithm for unimodal functions such as the golden section search algorithm [15], one can compute an accurate over-approximation of the minimal value of f and therefore of $\rho_{\mathcal{S}\cap\mathcal{G}}(\ell)$.

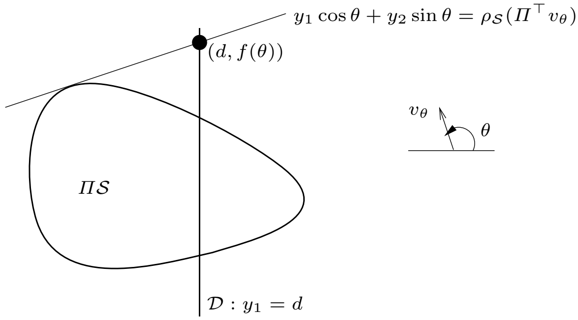


Fig. 3. Definition of the function $f(\theta)$

5 Support Functions Based Reachability Analysis

We now discuss effective reachability analysis in a location $q \in Q$ using support functions. For lighter notations, the index q has been dropped again. Let $\rho_{\mathcal{X}}$, $\rho_{\mathcal{U}}$, $\rho_{\mathcal{I}}$ denote the support functions of the sets $\mathcal{X}$, $\mathcal{U}$, $\mathcal{I}$; $\rho_{\mathcal{B}}$ denote the support function of the unit ball for the considered norm.

5.1 Over-Approximation of $\mathcal{R}_{\text{loc}}(q, \mathcal{X})$

We first determine a union of convex polyhedrons over-approximating $\mathcal{R}_{\text{loc}}(q, \mathcal{X})$.

Proposition 6. *Let $\mathcal{Z}_i$ be the sets defined in Algorithm 2, then*

$$\forall \ell \in \mathbb{R}^n,\ \rho_{\mathcal{Z}_i}(\ell) \leq \min\left(\rho_{\mathcal{Y}_i}(\ell), \min_{k=0}^{i} \rho_{\mathcal{I}_k}(\ell)\right),\ \textit{where}$$

$$\rho_{\mathcal{Y}_0}(\ell) = \max\left(\rho_{\mathcal{X}}(\ell), \rho_{\mathcal{X}}(e^{\tau A^\top}\ell) + \tau\rho_{\mathcal{U}}(\ell) + \alpha_\tau\rho_{\mathcal{B}}(\ell)\right),$$

$$\rho_{\mathcal{Y}_i}(\ell) = \rho_{\mathcal{Y}_0}(e^{i\tau A^\top}\ell) + \sum_{k=0}^{i-1}\left(\tau\rho_{\mathcal{U}}(e^{k\tau A^\top}\ell) + \beta_\tau\rho_{\mathcal{B}}(e^{k\tau A^\top}\ell)\right), \tag{3}$$

$$\rho_{\mathcal{I}_i}(\ell) = \rho_{\mathcal{I}}(e^{i\tau A^\top}\ell) + \sum_{k=0}^{i-1}\left(\tau\rho_{\mathcal{U}}(e^{k\tau A^\top}\ell) + \beta_\tau\rho_{\mathcal{B}}(e^{k\tau A^\top}\ell)\right). \tag{4}$$

Proof. For $i = 0$, by applying the rules of Proposition 4, it is straightforward to check that $\rho_{\mathcal{Z}_0}(\ell) \leq \min(\rho_{\mathcal{Y}_0}(\ell), \rho_{\mathcal{I}_0}(\ell))$. Hence, the property holds for $i = 0$. Let us assume that it holds for some i, then by definition of $\mathcal{Z}_{i+1}$ and from Proposition 4, we have $\rho_{\mathcal{Z}_{i+1}}(\ell) \leq \min(\rho_{\mathcal{Z}_i}(e^{\tau A^\top}\ell) + \tau\rho_{\mathcal{U}}(\ell) + \beta_\tau\rho_{\mathcal{B}}(\ell), \rho_{\mathcal{I}}(\ell))$. Then, by assumption

$$\rho_{\mathcal{Z}_i}(e^{\tau A^\top}\ell) + \tau\rho_{\mathcal{U}}(\ell) + \beta_\tau\rho_{\mathcal{B}}(\ell) \leq$$
$$\min\left(\rho_{\mathcal{Y}_i}(e^{\tau A^\top}\ell) + \tau\rho_{\mathcal{U}}(\ell) + \beta_\tau\rho_{\mathcal{B}}(\ell),\ \min_{k=0}^{i} \rho_{\mathcal{I}_k}(e^{\tau A^\top}\ell) + \tau\rho_{\mathcal{U}}(\ell) + \beta_\tau\rho_{\mathcal{B}}(\ell)\right)$$

Further, by equation (3)

$$\rho_{\mathcal{Y}_i}(e^{\tau A^\top}\ell) = \rho_{\mathcal{Y}_0}(e^{(i+1)\tau A^\top}\ell) + \sum_{k=0}^{i-1}\left(\tau\rho_{\mathcal{U}}(e^{(k+1)\tau A^\top}\ell) + \beta_\tau\rho_{\mathcal{B}}(e^{(k+1)\tau A^\top}\ell)\right)$$
$$= \rho_{\mathcal{Y}_0}(e^{(i+1)\tau A^\top}\ell) + \sum_{k=1}^{i}\left(\tau\rho_{\mathcal{U}}(e^{k\tau A^\top}\ell) + \beta_\tau\rho_{\mathcal{B}}(e^{k\tau A^\top}\ell)\right).$$

Therefore, it follows that $\rho_{\mathcal{Y}_i}(e^{\tau A^\top}\ell) + \tau\rho_{\mathcal{U}}(\ell) + \beta_\tau\rho_{\mathcal{B}}(\ell) = \rho_{\mathcal{Y}_{i+1}}(\ell)$. Similarly, we can show from equation (4) that $\rho_{\mathcal{I}_k}(e^{\tau A^\top}\ell) + \tau\rho_{\mathcal{U}}(\ell) + \beta_\tau\rho_{\mathcal{B}}(\ell) = \rho_{\mathcal{I}_{k+1}}(\ell)$. This leads to

$$\rho_{\mathcal{Z}_i}(e^{\tau A^\top}\ell) + \tau\rho_{\mathcal{U}}(\ell) + \beta_\tau\rho_{\mathcal{B}}(\ell) \le \min\left(\rho_{\mathcal{Y}_{i+1}}(\ell), \min_{k=0}^{i}\rho_{\mathcal{I}_{k+1}}(\ell)\right)$$

which implies that

$$\rho_{\mathcal{Z}_{i+1}}(\ell) \le \min\left(\rho_{\mathcal{Y}_{i+1}}(\ell), \min_{k=0}^{i}\rho_{\mathcal{I}_{k+1}}(\ell), \rho_{\mathcal{I}}(\ell)\right) = \min\left(\rho_{\mathcal{Y}_{i+1}}(\ell), \min_{k=0}^{i+1}\rho_{\mathcal{I}_k}(\ell)\right)$$

Hence, by induction, the proposition is proved. ■

It follows from the previous proposition that $\mathcal{Z}_i \subseteq \mathcal{Y}_i \cap \mathcal{I}_i \cap \cdots \cap \mathcal{I}_0$ where $\mathcal{Y}_i$, $\mathcal{I}_i, \ldots, \mathcal{I}_0$ are the convex sets determined by their support functions $\rho_{\mathcal{Y}_i}$, $\rho_{\mathcal{I}_i}, \ldots, \rho_{\mathcal{I}_0}$. Let us remark that the sets $\mathcal{Y}_i$ are actually the same than those in section 2 and thus give an over-approximation of the states reachable from $\mathcal{X}$ under the dynamics of the linear differential inclusion. The sets $\mathcal{I}_i, \ldots, \mathcal{I}_0$ allow us to take into account the constraint that the trajectories must remain in the invariant $\mathcal{I}$ during the evolution. We shall not discuss the efficient implementation of the evaluation of the support functions, this can be found for $\rho_{\mathcal{Y}_i}$, in [9]. A similar approach based on ideas from [7] can be used for the functions $\rho_{\mathcal{I}_i}, \ldots, \rho_{\mathcal{I}_0}$.

We can now compute polyhedral over-approximations $\overline{\mathcal{Z}}_i$ of the sets $\mathcal{Z}_i$ defined in Algorithm 2. Let $\ell_1, \ldots, \ell_r \in \mathbb{R}^n$ be a set of directions used for approximation. Let

$$\gamma_{i,j} = \min\left(\rho_{\mathcal{Y}_i}(\ell_j), \min_{k=0}^{i}\rho_{\mathcal{I}_k}(\ell_j)\right),\ j = 1, \ldots r.$$

Then, it follows from Propositions 3 and 6 that

$$\mathcal{Z}_i \subseteq \overline{\mathcal{Z}}_i = \{x \in R^n :\ \ell_j \cdot x \le \gamma_{i,j},\ j = 1, \ldots, r\}.$$

Then, Proposition 2 leads to the following result:

Theorem 1. *Let $\overline{\mathcal{Z}}_i$ be the polyhedrons defined above, let $i^* \in \mathbb{N}$ be the smallest index such that $\overline{\mathcal{Z}}_{i^*} = \emptyset$. Then, $\mathcal{R}_{\mathrm{loc}}(q, \mathcal{X}) \subseteq \overline{\mathcal{Z}}_0 \cup \cdots \cup \overline{\mathcal{Z}}_{i^*-1}$.*

The choice of the vectors $\ell_1, \ldots, \ell_r \in \mathbb{R}^n$ is important for the quality of approximation. If the invariant is a polyhedron $\mathcal{I} = \{x \in \mathbb{R}^d :\ c_j \cdot x \le d_j,\ j = 1, \ldots, m\}$

where $c_1, \ldots, c_m \in \mathbb{R}^n$ and $d_1, \ldots, d_m \in \mathbb{R}$, it is useful to include the vectors $c_1, \ldots, c_m$. This way, it is ensured that $\overline{\mathcal{Z}}_0 \cup \cdots \cup \overline{\mathcal{Z}}_{i^*-1} \subseteq \mathcal{I}$. Also, by considering vectors of the form $e^{-k\tau A^\top} c_1, \ldots, e^{-k\tau A^\top} c_m$, for some values of $k \in \{1, \ldots, i^* - 1\}$, the constraints imposed by the invariant on the reachable set at a given time step are also taken into account k time steps further.

5.2 Over-Approximation of $\mathcal{R}_{\mathbf{jump}}(e, \mathcal{X})$

Let $\mathcal{H}_e = \mathrm{CH}(\mathcal{R} \cap \mathcal{G}_e)$ be the set defined in Algorithm 2. Let us remark that $\mathcal{H}_e = \mathrm{CH}(\mathcal{H}_{e,0} \cup \cdots \cup \mathcal{H}_{e,i^*})$ where $\mathcal{H}_{e,i} = \mathcal{Z}_i \cap \mathcal{G}_e$.

Over-approximation of $\mathcal{H}_{e,i}$. If $\mathcal{G}_e$ is a compact convex set defined by its support function $\rho_{\mathcal{G}_e}$, let

$$\delta_{e,i,j} = \min\left(\rho_{\mathcal{G}_e}(\ell_j), \rho_{\mathcal{Y}_i}(\ell_j), \min_{k=0}^{i} \rho_{\mathcal{I}_k}(\ell_j)\right),\ j = 1, \ldots r.$$

Then, it follows from Propositions 3 and 6 that

$$\mathcal{H}_{e,i} \subseteq \overline{\mathcal{H}}_{e,i} = \{x \in R^n :\ \ell_j \cdot x \leq \delta_{e,i,j},\ j = 1, \ldots, r\}.$$

If $\mathcal{G}_e$ is a hyperplane we can use the method presented in section 4.2 to compute a more accurate over-approximation of $\mathcal{H}_{e,i}$. First of all, let us remark that the over-approximation of the support function $\rho_{\mathcal{Z}_i}$ given by Proposition 6 is possibly non-convex. Then, it cannot be used to compute an over-approximation of $\rho_{\mathcal{Z}_i \cap \mathcal{G}_e}$ by the method explained in section 4.2 as the function to minimize might not be unimodal. However, from Proposition 6, it follows that

$$\mathcal{H}_{e,i} \subseteq (\mathcal{Y}_i \cap \mathcal{I}_0 \cap \cdots \cap \mathcal{I}_i) \cap \mathcal{G}_e = (\mathcal{Y}_i \cap \mathcal{G}_e) \cap (\mathcal{I}_0 \cap \mathcal{G}_e) \cap \cdots \cap (\mathcal{I}_i \cap \mathcal{G}_e). \quad (5)$$

We can check the emptiness of $\mathcal{Y}_i \cap \mathcal{G}_e$, $\mathcal{I}_0 \cap \mathcal{G}_e$,..., $\mathcal{I}_i \cap \mathcal{G}_e$ using the simple test described in section 4.2. If one of these sets is empty, then $\mathcal{H}_{e,i} \subseteq \overline{\mathcal{H}}_{e,i} = \emptyset$. Otherwise, let

$$\delta_{e,i,j} = \min\left(\rho_{\mathcal{Y}_i \cap \mathcal{G}_e}(\ell_j), \min_{k=0}^{i} \rho_{\mathcal{I}_k \cap \mathcal{G}_e}(\ell_j)\right),\ j = 1, \ldots r$$

where the support functions $\rho_{\mathcal{Y}_i \cap \mathcal{G}_e}, \rho_{\mathcal{I}_0 \cap \mathcal{G}_e}, \ldots, \rho_{\mathcal{I}_i \cap \mathcal{G}_e}$ can be computed by the method explained in section 4.2. Then, from Proposition 3 and equation (5), it follows that

$$\mathcal{H}_{e,i} \subseteq \overline{\mathcal{H}}_{e,i} = \{x \in R^n :\ \ell_j \cdot x \leq \delta_{e,i,j},\ j = 1, \ldots, r\}.$$

Over-approximation of $\mathcal{H}_e$. Let $\overline{\mathcal{H}}_{e,0} \ldots \overline{\mathcal{H}}_{e,i^*}$ be computed by one of the two methods described in the previous paragraph, let $I = \{i :\ \overline{\mathcal{H}}_{e,i} \neq \emptyset\}$. Let $\delta_{e,j} = \max_{i \in I} \delta_{e,i,j}$, then we have

$$\mathcal{H}_e \subseteq \mathrm{CH}(\overline{\mathcal{H}}_{e,0} \cup \cdots \cup \overline{\mathcal{H}}_{e,i^*}) \subseteq \overline{\mathcal{H}}_e = \{x \in R^n :\ \ell_j \cdot x \leq \delta_{e,j},\ j = 1, \ldots, r\}.$$

Proposition 2 leads to the following result:

Theorem 2. *Let* $\overline{\mathcal{H}}_e$ *be the polyhedron defined above, let* $\overline{\mathcal{X}}_e = C_e\overline{\mathcal{H}}_e \oplus \mathcal{V}_e$*. Then,* $\mathcal{R}_{\text{jump}}(e, \mathcal{X}) \subseteq \overline{\mathcal{X}}_e$.

Let us remark that $\overline{\mathcal{X}}_e$ need not be effectively computed as it can be represented by its support function $\rho_{\overline{\mathcal{X}}_e}(\ell) = \rho_{\overline{\mathcal{H}}_e}(C_e^\top \ell) + \rho_{\mathcal{V}_e}(\ell)$.

6 Examples

In this section, we show the effectiveness of our approach on some examples. All computations were performed on a Pentium IV, 3.2 GHz with 1 GB RAM.

5-dimensional benchmark. We propose to evaluate the over-approximation due to our way of handling hybrid dynamics. For that purpose, we consider the 5-dimensional linear differential inclusion from [6]. This artificial system was generated from a block diagonal matrix and a random change of variables. The initial set $\mathcal{X}_0$ is a cube of side 0.05 centered at $(1, 0, 0, 0, 0)^\top$ and the set of inputs $\mathcal{U}$ is a ball of radius 0.01 centered at the origin.

By introducing a switching hyperplane, we artificially build a hybrid system that has the same set of reachable continuous states. The hybrid system has two locations 1 and 2 and one discrete transition $(1, 2)$. The continuous dynamics is the same in each location, given by the 5-dimensional linear differential inclusion. The invariants are $\mathcal{I}_1 = \{x \in \mathbb{R}^5 : c \cdot x \leq d\}$ where $c \in \mathbb{R}^5$ and $d \in \mathbb{R}$, and $\mathcal{I}_2 = \mathbb{R}^5$. The guard $\mathcal{G}_{(1,2)} = \{x \in \mathbb{R}^5 : c \cdot x = d\}$ and the reset map $R_{(1,2)}$ is the identity map. We assume that the initial location is 1. We computed the reachable sets of the linear differential inclusion and of the hybrid system over 800 time steps $\tau = 0.005$. Their projection on the first two continuous variables are shown in Figure 4, for two different choices of c and d.

We can see that hybrid dynamics introduces an additional over-approximation, especially for the second system where the reachable set intersects the guard almost tangentially. The accuracy can be improved in two ways, we can reduce the time step and we can consider more vectors for computing the polyhedral

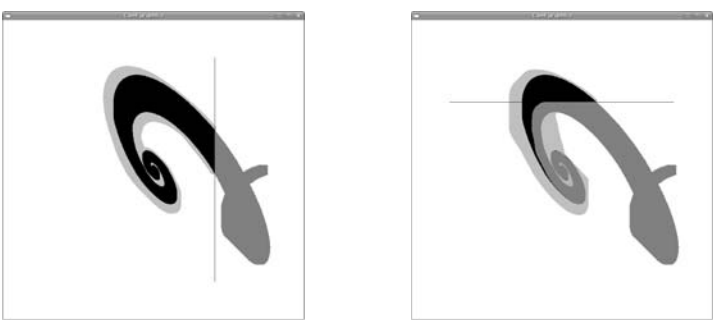


Fig. 4. Reachable set of the hybrid system in the first (dark grey), and second (light grey) locations, and reachable set of the equivalent linear differential inclusion (in black)

over-approximations. However, as we use a convex hull over-approximation of the intersection of the reachable set in the first location with the guard (see Algorithm 2), we will not reach the accuracy of the reachable set of the linear differential inclusion.

Navigation benchmark. We now consider the navigation benchmark for hybrid systems verification proposed in [10]. It models an object moving in a plane, whose position and velocities are denoted $x(t)$ and $v(t)$. The plane is partitioned into cubes, each cube corresponds to one location of the hybrid system. At time t, a location q is active if $x(t)$ is in the associated cube; there, the object follows dynamically a desired velocity $v_q \in \mathbb{R}^2$. We use the instances NAV01 and NAV04 from [16]. We render the problem slightly more challenging by including an additional continuous input $u(t)$ modelling disturbances. In the location q, the four-dimensional continuous dynamics is given by

$$\dot{x}(t) = v(t),\ \dot{v}(t) = A(v(t) - v_q - u(t)),\ \|u(t)\|_2 \leq 0.1 \text{ where } A = \left(\begin{smallmatrix} -1.2 & 0.1 \\ 0.1 & -1.2 \end{smallmatrix}\right).$$

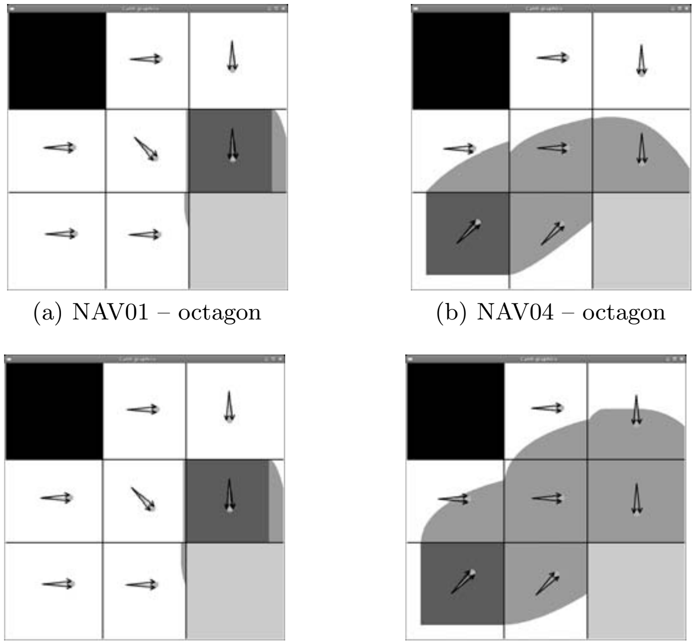
In Figure 5, we represented the projection of the reachable sets on the position variables as well as the partition of the plane and, in each cube of the partition the set of velocities $v_q \oplus 0.1\mathcal{B}$ in the associated location.

One can check in Figure 5(b) that the intersection of the over-approximation $\overline{\mathcal{Z}}_i$ of the reachable sets in one location with the guards does not coincide with the over-approximation $\overline{\mathcal{H}}_e$ of the intersection of the reachable set in one location with the guards. The latter is more accurate because of the use of the method presented in section 4.2 for computing the support function of the intersection of a convex set with a hyperplane.

In Figures 5(c) and 5(d), the support function of the intersection is sampled in 6 directions which results in $\overline{\mathcal{H}}_e$ defined as an interval hull. We need to add 12 sampling directions to get an octagon, used in Figures 5(a) and 5(b). The benefit of using more sampling directions for approximation is clearly seen for NAV04 where the reachable set appears to be actually much smaller in Figure 5(b) than in Figure 5(d). However, the support function of an interval hull can be computed very efficiently whereas the support function of an octagon requires solving a linear program. This explains the huge differences in execution times reported in Table 1.

Table 1 also contains time and memory used by the optimized tool PHAVer on a similar computer as reported in [16]. One should be careful when comparing these results. On one hand the navigation problem we consider here is more challenging than the original one since we add disturbances on the input. These disturbances add, in several locations, chattering effects that cannot occur without them, and produces larger reachable sets. On the other hand PHAVer uses exact arithmetic.

We believe that the performances of our algorithm can be significantly improved. First of all, our algorithm is slowed down due to an inefficient interface between our reachability algorithm and the LP solver used to evaluate support functions. Also, there is a lot of room for improvements in the implementation. A first improvement would be to initialize the LP solver with the last computed



(a) NAV01 – octagon (b) NAV04 – octagon

(c) NAV01 – interval hull (d) NAV04 – interval hull

Fig. 5. Two navigation benchmarks with perturbations. In (a) and (b) the intersections with the guards are over-approximated by octagons, whereas in (c) and (d) they are over-approximated by their interval hulls. Dark grey: initial set. Grey: reachable sets. Light grey: Target State. Black: Forbidden State.

Table 1. Time and memory needed for reachability analysis of NAV01 and NAV04

	NAV01		NAV04	
	time (s)	memory (MB)	time (s)	memory (MB)
octagon	10.28	0.24	54.77	0.47
interval hull	0.11	0.24	0.88	0.47
PHAVer	8.7	29.0	13.6	47.6

optimizer, because ℓ and $e^{\tau A^\top}\ell$ are almost the same. Another improvement would be to over-approximate the intersections with the guards by several sets whose support function have growing complexity in order to avoid calling the LP solver as much as possible.

7 Conclusion

In this paper we presented a new method for conservative reachability analysis of a class of hybrid systems. The use of support functions allows us to consider a wide class of input sets, invariants, and guards. For the special case of guards defined by hyperplanes we showed how to transform the problem of intersecting the reachable set with a guard to the minimization of a unimodal function. Our algorithms have been implemented in a prototype tool that shows promising results on non-trivial examples. There is still a lot of room for improvements, future work should focus on the choice of the directions of approximation and LP optimizations.

References

1. Chutinan, A., Krogh, B.: Verification of polyhedral-invariant hybrid automata using polygonal flow pipe approximations. In: Vaandrager, F.W., van Schuppen, J.H. (eds.) HSCC 1999. LNCS, vol. 1569, pp. 76–90. Springer, Heidelberg (1999)
2. Asarin, E., Dang, T., Maler, O., Bournez, O.: Approximate reachability analysis of piecewise-linear dynamical systems. In: Lynch, N.A., Krogh, B.H. (eds.) HSCC 2000. LNCS, vol. 1790, pp. 20–31. Springer, Heidelberg (2000)
3. Kurzhanski, A., Varaiya, P.: Ellipsoidal techniques for reachability analysis. In: Lynch, N.A., Krogh, B.H. (eds.) HSCC 2000. LNCS, vol. 1790, pp. 202–214. Springer, Heidelberg (2000)
4. Botchkarev, O., Tripakis, S.: Verification of hybrid systems with linear differential inclusions using ellipsoidal approximations. In: Lynch, N.A., Krogh, B.H. (eds.) HSCC 2000. LNCS, vol. 1790, pp. 73–88. Springer, Heidelberg (2000)
5. Stursberg, O., Krogh, B.: Efficient representation and computation of reachable sets for hybrid systems. In: Maler, O., Pnueli, A. (eds.) HSCC 2003. LNCS, vol. 2623, pp. 482–497. Springer, Heidelberg (2003)
6. Girard, A.: Reachability of uncertain linear systems using zonotopes. In: Morari, M., Thiele, L. (eds.) HSCC 2005. LNCS, vol. 3414, pp. 291–305. Springer, Heidelberg (2005)
7. Girard, A., Le Guernic, C., Maler, O.: Efficient computation of reachable sets of linear time-invariant systems with inputs. In: Hespanha, J.P., Tiwari, A. (eds.) HSCC 2006. LNCS, vol. 3927, pp. 257–271. Springer, Heidelberg (2006)
8. Varaiya, P.: Reach set computation using optimal control. In: Proc. KIT Workshop on Verification of Hybrid Systems, Grenoble (1998)
9. Le Guernic, C., Girard, A.: Reachability analysis of linear systems using support functions. Nonlinear Analysis: Hybrid Systems (2009) (to appear)
10. Fehnker, A., Ivancic, F.: Benchmarks for hybrid systems verification. In: Alur, R., Pappas, G.J. (eds.) HSCC 2004. LNCS, vol. 2993, pp. 326–341. Springer, Heidelberg (2004)
11. Bertsekas, D., Nedic, A., Ozdaglar, A.: Convex analysis and optimization. Athena Scientific (2003)
12. Boyd, S., Vandenberghe, L.: Convex optimization. Cambridge University Press, Cambridge (2004)
13. Rockafellar, R., Wets, R.: Variational analysis. Springer, Heidelberg (1998)
14. Girard, A., Le Guernic, C.: Zonotope/hyperplane intersection for hybrid systems reachability analysis. In: Egerstedt, M., Mishra, B. (eds.) HSCC 2008. LNCS, vol. 4981, pp. 215–228. Springer, Heidelberg (2008)
15. Kiefer, J.: Sequential minimax search for a maximum. Proceedings of the American Mathematical Society 4, 502–506 (1953)
16. Frehse, G.: PHAVer: algorithmic verification of hybrid systems past HyTech. Int. J. on Software Tools for Technology Transfer 10(3), 263–279 (2008)