Tracking Smooth Trajectories in Linear Hybrid Systems[☆]

Massimo Benerecetti, Marco Faella*

Dept. of Electrical Engineering and Information Technology
Università di Napoli "Federico II"
Via Claudio 21, 80125, Napoli, Italy

Abstract

We analyze the properties of smooth trajectories subject to a constant differential inclusion which constrains the first derivative to belong to a given convex polyhedron. We present the first exact symbolic algorithm that computes the set of points from which there is a trajectory that reaches a given polyhedron while avoiding another (possibly non-convex) polyhedron. We prove that this set of points remains the same if the smoothness constraint is replaced by a weaker differentiability constraint, but not if it is replaced by almost everywhere differentiability. We discuss the connection with (Linear) Hybrid Automata and in particular the relationship with the classical algorithm for reachability analysis for Linear Hybrid Automata.

Keywords: hybrid automata, reachability, controller synthesis

1. Introduction

Hybrid Automata are a mathematical abstraction of systems that feature both discrete and continuous dynamics. Linear Hybrid Automata (LHAs) [1] were introduced as a computationally tractable model of hybrid systems that still allows for non-trivial dynamics. In particular, LHAs can approximate complex dynamics up to an arbitrary precision [2].

In an LHA, discrete dynamics is represented by a finite set of control modes called *locations*, while the continuous dynamics is embodied by a finite set of real-valued variables. In each location, the continuous dynamics is constrained by a differential inclusion of the type $\dot{x} \in F$, where $\dot{x}$ is the vector of the time-derivatives of all the variables in the system, and $F \subseteq \mathbb{R}^n$ is a convex polyhedron. The main decision problem that was considered for LHAs is *reachability*, i.e., given two system configurations, say an initial state and an error state, establish whether there is a system behavior that leads from the first to the second. A more complex task consists in verifying whether a given LHA can be modified (i.e., *controlled*) in such a way that a given error configuration (or region) is *not* reached by any behavior. This problem can be called *safety control* and is analogous to a game with a safety objective. Both problems require an algorithm for the following sub-problem, which applies to a single discrete location: given a region G (for *goal*) and a region A (for *avoid*) of system configurations, find the set of points from which there is a trajectory that reaches G while avoiding A at all times. We denote this set by $RWA(G, A)$ for *reach while avoiding*. In reachability problems, the goal region G can be thought of as comprising error states, and the avoidance region A is the complement of the *invariant* of the automaton, which is the set of configurations that make physical sense for the system. Hence, $RWA(G, A)$ is the set of states that reach an error state while remaining in the invariant. In a safety control problem, the goal region G is taken to be a set of uncontrollable states (such as, states outside the safe region) and A is a set of controllable states (included in the invariant). Then,

[☆]This work was partially supported by the INdAM-GNCS project "Logica e Automi per il Model-Checking Intervallare".
*Corresponding author.
Email addresses: massimo.benerecetti@unina.it (Massimo Benerecetti), m.faella@unina.it (Marco Faella)

$RWA(G, A)$ identifies the region in which the environment can reach an error state while avoiding the good, controllable states.

The *RWA* operator is recognized as a central tool in the analysis of various kinds of hybrid systems: it corresponds to the *Reach* operator in Tomlin et al. [3] and *Unavoid_Pre* in Balluchi et al. [4]; it was also used in the synthesis of controllers for reachability objectives [5, 6].

Computing reach-while-avoiding. The algorithmic computation of $RWA(G, A)$ is simple when A is co-convex, corresponding to the case of reachability analysis for LHAs with convex invariants. In that case, *RWA* can be expressed in the first-order theory of reals and computed using a constant number of basic polyhedra operations.

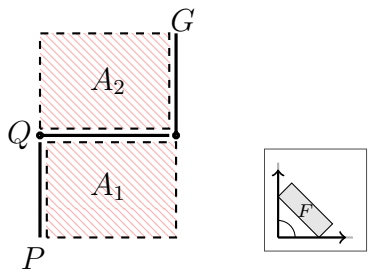


Figure 1: Can the points in P reach the target region G while remaining in $P \cup Q \cup G$? The flow constraint F is shown on the r.h.s.

When A is not co-convex, one may adapt the procedure that is presented in one of the early papers on LHAs, in the context of reachability analysis in presence of non-convex invariants [1]. The idea of the algorithm is simple: consider a partition of the non-convex invariant I into a finite set of convex polyhedra $P_1, \ldots, P_n$; then, split the location with invariant I into n different locations, each with convex invariant P_i; finally, connect these new locations with virtual transitions corresponding to the boundaries between two adjacent convex polyhedra P_i, P_j. Because of the added virtual transitions, this approach naturally leads to trajectories that are *almost everywhere (a.e.) differentiable*.

Consider for example the situation depicted in Figure 1. Assume that the invariant for the current location is $P \cup Q \cup G$ and the goal is to reach G. Dashed lines identify topologically open sides of polyhedra. The flow constraint F is also depicted in the figure: it allows trajectories to move in a range of directions going from straight right to straight up, and it forbids stopping (i.e., it does not include the origin).

The above procedure splits the invariant into three convex polyhedra, and then performs a backward reachability analysis which starts from the goal G and progressively enlarges the set of "good" states W by including the states that can reach W while remaining in one of the convex parts of the invariant.

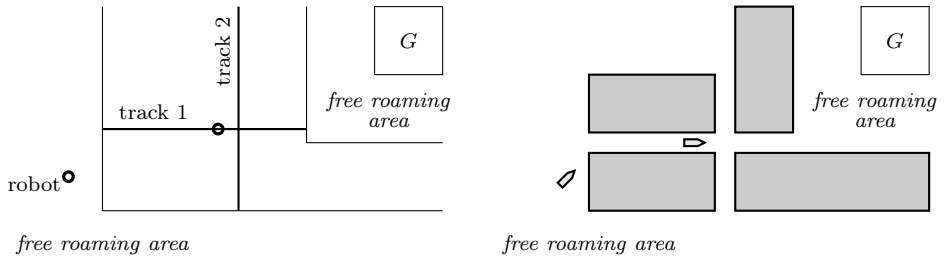
In our example, the points in the line segment Q can reach the target by moving straight to the right, while remaining in one convex part of the invariant and, similarly, points in the line segment P can reach the extreme point of Q by moving straight up. Hence, both Q and P end up on the final solution. On the other hand, no differentiable trajectory can start in a point of P and reach G while remaining in $P \cup Q \cup G$. [1]

In some scenarios, such as the one we describe below, restricting the system trajectories to be differentiable, or even smooth (i.e., differentiable an arbitrary number of times at all times), may be desirable to ensure that certain physical constraints are satisfied. In this paper, we present an exact algorithm for computing $RWA(G, A)$ for general polyhedra G and A with respect to smooth trajectories. Moreover, we prove that differentiable and smooth trajectories lead to equivalent notions of *RWA*.

Applications. The difference between RWA under smooth trajectories and under a.e. differentiable ones for LHAs only surfaces when the avoidance region A is not topologically closed. To see how this case may be relevant to applications, consider the example in Figure 2(a), where multiple robots (such as Kiva Systems[2]) must visit a target region G at the same time. The robots can move between two free roaming areas connected by two linear, intersecting tracks, and, for safety reasons, must always keep a minimum distance between each other. In this scenario, a topologically open avoidance region can be used to model the two intersecting tracks, as shown in Figure 3. Requiring smoothness (or even plain differentiability) of the allowed trajectories

[1] By rotating the polyhedra in Figure 1 (including the flow constraint F) by 45°, it becomes apparent that the issue also occurs with *rectangular* flow constraints. However, Rectangular Hybrid Automata and Games [7] do not exhibit the above issue, due to the presence of multiple restrictions, such as the fact that guards and invariants are convex and topologically closed.

[2] A commercial robotic platform for warehouse automation: `http://www.kivasystems.com`.



(a) The robots scenario

(b) The ships scenario

Figure 2: Two scenarios in which a target region G must be reached while moving in two free roaming areas connected by narrow tracks (a) or channels (b).

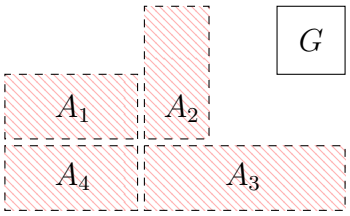


Figure 3: Target area G and avoidance region $A = \bigcup_{i=1}^{4} A_i$ for the two application scenarios depicted in Figure 2. The polyhedra $A_1, \ldots, A_4$ are *weakly adjacent*, i.e., their topological closures intersect.

ensures that robots cannot switch track at the intersection. The process of moving along a track could also be abstracted away by a discrete mode change (i.e., robots would seem to jump from one end of the track to the other end), but this prevents the explicit modeling of some phenomena, such as the possible collisions between two robots traveling on different tracks.

Figure 2(b) shows a different interpretation of the same scenario, in which some ships have to reach the target region G, using two narrow intersecting channels. The small width of the channels compared to the size of the ships prevents the ships from turning at any point, including the point where the channels intersect. Once again, the topologically open avoidance region shown in Figure 3 is a convenient modeling technique for this scenario.

Structure of the paper. The rest of the paper is organized as follows. Section 2 is devoted to preliminary definitions, including the problem statement and the known algorithm for computing *RWA* with respect to a.e. differentiable trajectories. In Section 3 we show that differentiable trajectories are equivalent to smooth trajectories for our purposes. Sections 4 and 5 present the main result of the paper, namely a procedure to compute the *RWA* operator for differentiable trajectories. This is done by first introducing in Section 4 the notion of *type* of a trajectory as the sequence of polyhedra traversed by the trajectory in a given time interval. Then, in Section 5, a recursive procedure to compute the points having a given type is proposed, whose effective computability is relative to the computability of two operators, Ext_2 and Ext_3. Being the set of all types finite, this provides an algorithm for *RWA*. The rest of that section shows how those two operators can effectively be computed by means of symbolic operations on polyhedra. While the case of Ext_2 is relatively simple, the computation of Ext_3 is much more involved and requires non-trivial sequence of polyhedra operations. Section 6 provides a alternative fixpoint characterization of the *RWA*, based on the same operators Ext_2 and Ext_3, that is more suitable to an implementation. The results of experiments, comparing an implementation of the algorithm proposed in the paper with the one dealing with a.e. differentiable trajectories, are described in Section 7. Finally, we provide some conclusions in Section 8.

The present paper extends and improves a preliminary version [8], which did not take into account smooth trajectories and did not contain detailed proofs of all claims. Moreover, here we develop the classification of witnesses in types and propose and implement an algorithm for *RWA* based on it. The experiment section is also novel.

2. Preliminaries and Problem Definition

Let $\mathbb{R}$ (respectively, $\mathbb{R}_{\geq 0}$) denote the set of real numbers (resp., non-negative real numbers). Throughout the paper we consider a fixed ambient space $\mathbb{R}^n$. A *convex polyhedron* is a subset of $\mathbb{R}^n$ that is the intersection of a finite number of open and closed half-spaces. A *polyhedron* is a subset of $\mathbb{R}^n$ that is the union of a finite number of convex polyhedra. For a general (i.e., not necessarily convex) polyhedron $G \subseteq \mathbb{R}^n$, we denote by $cl(G)$ its topological closure, by $\overline{G}$ its complement, and by $[\![G]\!] \subseteq 2^{\mathbb{R}^n}$ its representation as a finite set of convex polyhedra. We assume w.l.o.g. that $[\![G]\!]$ contains mutually disjoint convex polyhedra, called *patches* of G.

Let $\mathcal{C} \subseteq [\mathbb{R}_{\geq 0} \to \mathbb{R}^n]$ be a class of functions from the time domain to our ambient space. Given a convex polyhedron F, called the *flow constraint*, an $(F, \mathcal{C})$-*trajectory* is a function $f \in \mathcal{C}$ such that $\dot{f}(t) \in F$ for all $t \geq 0$ such that f is differentiable in t. Given a point $x \in \mathbb{R}^n$, let $Adm_F^{\mathcal{C}}(x)$ (for *admissible*) denote the set of $(F, \mathcal{C})$-trajectories f starting from x (i.e., such that $f(0) = x$). We henceforth consider three classes of functions: the class $\mathcal{C}_{\mathrm{s}}$ of functions that are smooth everywhere, the class $\mathcal{C}_{\mathrm{d}}$ of functions that are continuous and differentiable everywhere, and the class $\mathcal{C}_{\mathrm{ae}}$ of functions that are continuous everywhere and differentiable almost everywhere (i.e., always except for a finite set of time points). Notice that $\mathcal{C}_{\mathrm{s}}$ corresponds to the differentiability class C^∞ of infinitely differentiable functions; however $\mathcal{C}_{\mathrm{d}}$ does not coincide with C^1, as it admits continuous functions whose first derivatives are not continuous.

Given two disjoint polyhedra G (for *goal*) and A (for *avoid*), we denote by $RWA_F^{\mathcal{C}}(G, A)$ (for *reach while avoiding*) the set of points from which there is an $(F, \mathcal{C})$-trajectory that reaches G while avoiding A. Formally, we have:

$$RWA_F^{\mathcal{C}}(G, A) = \Big\{ x \in \mathbb{R}^n \;\Big|\; \exists f \in Adm_F^{\mathcal{C}}(x),\, \delta_f \geq 0 : f(\delta_f) \in G \text{ and } \forall\, \delta \in [0, \delta_f] : f(\delta) \notin A \Big\}.$$

For every $x \in RWA_F^{\mathcal{C}}(G, A)$, any pair (f, δ_f) satisfying the condition in the definition of $RWA_F^{\mathcal{C}}$ will be called an $(F, \mathcal{C})$-*witness* for x, when G and A are clear from the context. Since clearly $\mathcal{C}_{\mathrm{s}} \subseteq \mathcal{C}_{\mathrm{d}} \subseteq \mathcal{C}_{\mathrm{ae}}$, we immediately obtain that $RWA_F^{\mathcal{C}_{\mathrm{s}}}(G, A) \subseteq RWA_F^{\mathcal{C}_{\mathrm{d}}}(G, A) \subseteq RWA_F^{\mathcal{C}_{\mathrm{ae}}}(G, A)$. We shall see in the following that the first inclusion is in fact an equivalence, whereas the second inclusion is strict. Notice also that

$$RWA_F^{\mathcal{C}}(G_1 \cup G_2, A) = RWA_F^{\mathcal{C}}(G_1, A) \cup RWA_F^{\mathcal{C}}(G_2, A),$$

whereas $RWA_F^{\mathcal{C}}$ does not distribute over unions in the second argument (see [9]). Therefore, in the following we assume w.l.o.g. that the goal G is a convex polyhedron.

We assume that we can compute the following basic operations on arbitrary convex polyhedra P and P': the Boolean operations $P \cup P'$, $P \cap P'$, and $\overline{P}$; the topological closure $cl(P)$ of P; finally, the *pre-* and *post-flows* of P, defined as follows:

$$P{\swarrow} = \{x - \delta c \mid x \in P, \delta \geq 0, c \in F\} \qquad P{\swarrow_{>0}} = \{x - \delta c \mid x \in P, \delta > 0, c \in F\}$$
$$P{\nearrow} = \{x + \delta c \mid x \in P, \delta \geq 0, c \in F\} \qquad P{\nearrow_{>0}} = \{x + \delta c \mid x \in P, \delta > 0, c \in F\}.$$

Intuitively, the pre- and post-flow operators compute the pre- and post-image, respectively, of a convex polyhedron with respect to the straight directions contained in F. The algorithm for $P{\swarrow_{>0}}$ and $P{\nearrow_{>0}}$ can be found in [10].

It is well known that $P{\swarrow}$ (resp., $P{\nearrow}$) is not a convex polyhedron when F is non-necessarily closed. The following example shows that the same is true even if both P and F are closed convex polyhedra. This observation contradicts a claim made by Halbwachs et al. [11].

Theorem 1. *Given two convex polyhedra P and F, the following hold:*

1. *$P\nearrow$ may not be a convex polyhedron, even if both P and F are closed;*
2. *$P\nearrow = P \cup (P\nearrow_{>0})$;*
3. *$P\nearrow \cap cl(P)$ is a convex polyhedron;*
4. *if $P \subseteq P\nearrow_{>0}$, then $P\nearrow$ is a convex polyhedron.*

PROOF. To show that the first property holds it suffices to consider the closed convex polyhedra $P = \{(0,0)\}$ and $F = \{(x,y) \mid x \geq 1\}$. According to the definition, $P\nearrow = P \cup \{(x,y) \mid x > 0\}$, which is a convex set but not a convex polyhedron. As to the second property, it is enough to observe that $P = \{x + \delta c \mid x \in P, \delta = 0, c \in F\}$.

To prove the third property it suffices to show that given $x \in P\nearrow \cap cl(P) \triangleq P'$ and $y \in cl(P') \subseteq cl(P)$, any strict convex combination of x and y belongs to P'.

Since $x \in P'$, then there must exist a point $u \in P$, a time $\delta \geq 0$ and a flow direction $c \in F$, such that $x = u + \delta\, c$. If $\delta = 0$ then $x \in P$ and the thesis follows from the polyhedricity of P and the fact that $y \in cl(P)$. Indeed, any strict convex combination of x and y belongs to P. Assume $\delta > 0$ and consider the triangle Q with vertices $u \in P$ and $x, y \in cl(P)$ (see Fig. 4). Clearly, all the points contained in the relative interior of Q belong to P. Let z be a strict convex combination of x and y, hence $z \in cl(P)$. It is easy to see that there exist a time $\delta' > 0$ and a point $u' = z - \delta' c$ such that u' belongs to the relative interior of Q and, therefore, it belongs to P as well.

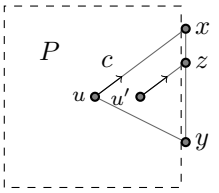


Figure 4: Proving that $P\nearrow \cap cl(P)$ is a convex polyhedron (see Theorem 1). The square P is topologically open, and x, y, z lie on its boundary.

For the final property, the assumption $P \subseteq P\nearrow_{>0}$ and Item 2 imply that $P\nearrow = P\nearrow_{>0}$. Since, in addition, $P\nearrow_{>0}$ is a convex polyhedron, the thesis follows. ∎

2.1. The Existing Algorithm for Almost Everywhere Differentiable Trajectories

Starting from this section, we consider a fixed flow constraint F, and we omit the F parameter from all notations.

Let us briefly recall the previous approach to the problem, which can be used to compute *RWA*, assuming that trajectories are differentiable everywhere except for a finite set of time points. Given two convex polyhedra P, P', let $Reach(P, P')$ be the set of points in P that can reach P' via a trajectory that remains within $P \cup P'$ at all times (until P' is reached). Formally,

$$Reach(P, P') = \left\{x \in P \,\middle|\, \exists f \in Adm^{\mathcal{C}_{\text{ae}}}(x), \delta \geq 0 : f(\delta) \in P' \text{ and } \forall \delta' \in [0, \delta) : f(\delta') \in P \cup P'\right\}.$$

It can be shown that $Reach(P, P')$ is a convex polyhedron which can be computed from P, P', and F using basic operations on polyhedra [12, 9]. Then, Theorem 2 shows how $RWA^{\mathcal{C}_{\text{ae}}}$ can be computed by iterative application of *Reach*.

Theorem 2. *[12, 9] For all disjoint polyhedra G and A, and for all polyhedra W, let*

$$\tau_{\text{ae}}(G, A, W) = G \cup \bigcup_{P \in [\![\overline{A}]\!]} \bigcup_{P' \in [\![W]\!]} Reach(P, P').$$

We have $RWA^{\mathcal{C}_{\text{ae}}}(G, A) = \mu W\,.\,\tau_{\text{ae}}(G, A, W)$, where μW denotes the least fixed point. Moreover, the fixed point is reached within a finite number of iterations.

By Knaster-Tarski fixed point theorem, the fixed point equation in Theorem 2 suggests the semi-algorithm consisting in repeated applications of $\tau_{\text{ae}}(G, A, W)$, starting from $W = \emptyset$, i.e.: $W_0 = \emptyset$ and $W_{i+1} = \tau_{\text{ae}}(G, A, W_i)$ for all $i \geq 0$. Theorem 2 states that there exists $k > 0$ such that $W_k = W_{k+1} = RWA^{\mathcal{C}_{\text{ae}}}(G, A)$.

The *Reach* and τ_{ae} operators, together with Theorem 2, represent a reformulation of the original algorithm for reachability analysis of LHAs under non-convex invariants, which was expressed in terms of locations of a hybrid automaton. Notice that both ourselves (in [12], later corrected by [9]) and Alur et al. (in [13, 1]) have claimed that τ_{ae} (or very similar variations thereof) can be used to compute $RWA^{\mathcal{C}_{\text{d}}}$. Those claims are incorrect, as shown in the Introduction and again below.

Adapting the known algorithm for $RWA^{\mathcal{C}_{\text{ae}}}$ to differentiable trajectories is not a trivial task, as the following observations show. Indeed, one may try to modify the definition of *Reach*, by simply replacing $Adm^{\mathcal{C}_{\text{ae}}}$ with $Adm^{\mathcal{C}_{\text{d}}}$. However, this replacement is not sufficient to solve the problem with the example in Figure 1: it would still hold $Reach(P, Q) = P$, because the points in P can reach Q along a straight-line trajectory, which is both in $\mathcal{C}_{\text{ae}}$ and in $\mathcal{C}_{\text{d}}$. A somewhat deeper modification might be attempted, after noticing that all trajectories going from P to Q lie within P at all times, except for the final point, which belongs to Q. Hence, one could modify the definition of $Reach(P, P')$, by requiring not only that there exists a (differentiable) trajectory from P to P' contained in $P \cup P'$, but also that this trajectory *spends a positive amount of time in* P':

$$\begin{aligned} Reach'(P, P') = \big\{ x \in P \,\big|\, \exists f \in Adm^{\mathcal{C}_{\text{d}}}(x), 0 \leq \delta_1 < \delta_2 : \\ \forall \delta \in (0, \delta_1] : f(\delta) \in P \cup P' \text{ and } \forall \delta \in (\delta_1, \delta_2] : f(\delta) \in P' \big\}. \end{aligned}$$

Unfortunately, $Reach'$ still suffers from a shortcoming. Consider again the example in Figure 1, but this time let the avoidance region be $A = A_1$. Notice that the status of the immediate neighborhood of P is identical to the previous case: Q is still a "good" neighbor (w.r.t. reaching G while avoiding A) and A_1 is still a "bad" neighbor. However, we have $P \subseteq RWA^{\mathcal{C}_{\text{d}}}(G, A)$, because a differentiable trajectory can start from P, pass instantaneously through the left vertex of Q, then curve into A_2 and finally reach G. The fact that P is a set of good points is essentially due to A_2, which is not an immediate neighbor of P (we may call it a *weak* neighbor, since its topological closure intersects the one of P, i.e., $cl(P) \cap cl(A_2) \neq \emptyset$).

Therefore, we realize that $Reach'$ cannot solve this example because it is constrained to consider pairs of adjacent convex polyhedra. In particular, it holds $Reach'(P, A_2) = \emptyset$ because P and A_2 are not adjacent, and $Reach'(P, Q) = \emptyset$ because differentiable trajectories cannot start from P and spend a positive amount of time in Q while remaining in $P \cup Q$.

In the rest of the paper we show that significant new developments are required to correctly compute $RWA^{\mathcal{C}_{\text{d}}}$ and $RWA^{\mathcal{C}_{\text{s}}}$.

3. Equivalence of Differentiable and Smooth Trajectories

The examples discussed above imply that the inclusion between $RWA^{\mathcal{C}_{\text{d}}}(G, A)$ and $RWA^{\mathcal{C}_{\text{ae}}}(G, A)$ is strict in general, i.e., $RWA^{\mathcal{C}_{\text{d}}}(G, A) \subsetneq RWA^{\mathcal{C}_{\text{ae}}}(G, A)$. On the contrary, we can show that $RWA^{\mathcal{C}_{\text{d}}}(G, A)$ and $RWA^{\mathcal{C}_{\text{s}}}(G, A)$ coincide for all G and A. To do so, we shall need to prove two technical lemmas first. Consider a differentiable trajectory lying within a convex polyhedron P. Along any tangent to the curve one can find a point that is still in the closure of P (if not in P itself) and that is reachable from the initial point of the curve. The following lemma formalizes this property and is proved in the Appendix.

Lemma 1 (Tangent). *Let P be a convex polyhedron, x a point, $f \in Adm^{\mathcal{C}_{\text{d}}}(x)$ a trajectory, and $\hat{\delta} > 0$ a delay such that in all non-empty intervals $(\delta, \hat{\delta})$ there is a time γ such that $f(\gamma) \in P$. Then, there exists $\delta^* > 0$ such that $f(\hat{\delta}) - \delta^* \dot{f}(\hat{\delta}) \in cl(P) \cap \{x\} \nearrow_{>0}$.*

The following lemma, illustrated in Figure 5, shows how to connect, in an admissible and smooth fashion, two points which can be connected by the concatenation of two straight-line admissible trajectories.

Lemma 2 (Smooth Interpolation). *Given three points x_0, x_1, $x_2 \in \mathbb{R}^n$, two directions $c_0, c_1 \in F$ and two delays $\delta_0, \delta_1 \geq 0$ such that: $x_1 = x_0 + \delta_0 c_0$ and $x_2 = x_1 + \delta_1 c_1$, there exists a trajectory $f \in Adm^{\mathcal{C}_{\text{d}}}(x_0)$ such that $f(\delta_0 + \delta_1) = x_2$ and:*

(i) f is smooth in the interval $[0, \delta_0 + \delta_1]$;

(ii) $\dot{f}(0) = c_0$ *and* $\dot{f}(\delta_0 + \delta_1) = c_1$*;*

(iii) all higher-order derivatives $f^{(k)}$*, with* $k > 1$*, are such that* $f^{(k)}(0) = 0$ *and* $f^{(k)}(\delta_0 + \delta_1) = 0$*;*

(iv) $f(\delta)$ *is a strict convex combination of* x_0, x_1, x_2 *for all* $\delta \in (0, \delta_0 + \delta_1)$*.*

PROOF. We assume for simplicity that x_0, x_1, x_2 are not collinear and hence identify a unique plane. The argument for collinear points requires only minor modifications.

First, we make sure that the first derivative of f is admissible (i.e., it lies in F), by stipulating that it is a convex combination of c_0 and c_1. Let $\bar{\delta} = \delta_0 + \delta_1$, we set:

$$\dot{f}(\delta) = (1 - g(\delta))c_0 + g(\delta)c_1 \qquad \text{for all } \delta \in [0, \bar{\delta}],$$

for a suitable function $g : [0, \bar{\delta}] \to [0, 1]$. As a consequence, we obtain that:

$$\begin{aligned} f(\delta) &= \int_0^\delta g(\gamma)c_1 + \big(1 - g(\gamma)\big)c_0 d\gamma + C \\ &= \int_0^\delta g(\gamma)c_1 + c_0 - g(\gamma)c_0 d\gamma + C \\ &= \delta c_0 + (c_1 - c_0) \int_0^\delta g(\gamma) d\gamma + C. \end{aligned}$$

Recall that our thesis requires the following constraints to be true.

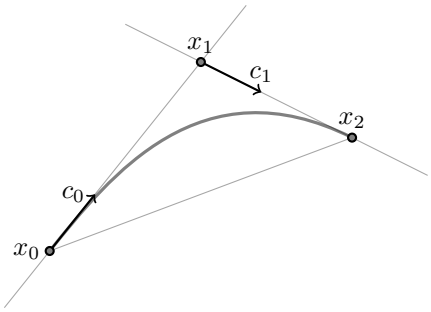


Figure 5: Connecting two intersecting lines in an admissible and differentiable way (see Lemma 2).

$$\begin{aligned} f(0) &= x_0 & f(\bar{\delta}) &= x_2 = x_0 + \delta_0 c_0 + \delta_1 c_1 \\ \dot{f}(0) &= c_0 & \dot{f}(\bar{\delta}) &= c_1 \\ f^{(k)}(0) &= f^{(k)}(\bar{\delta}) = 0 \qquad \text{for all } k > 1. \end{aligned}$$

As a consequence, we obtain that $C = x_0$ and the function g must satisfy the following conditions:

$$g(0) = 0 \tag{1}$$
$$g(\bar{\delta}) = 1 \tag{2}$$
$$g(\delta) \in [0, 1] \qquad \text{for all } \delta \in [0, \bar{\delta}] \tag{3}$$
$$g^{(k)}(0) = g^{(k)}(\bar{\delta}) = 0 \qquad \text{for all } k \geq 1 \tag{4}$$
$$\int_0^{\bar{\delta}} g(\gamma) d\gamma = \delta_1. \tag{5}$$

Consider the logistic function $LF(\delta)$, defined as: $\frac{1}{1+e^{-\delta}}$. As function $g(\delta)$, we choose the following:

$$g(\delta) = \begin{cases} 0 & \text{if } \delta = 0 \\ LF(h(\delta)) & \text{if } \delta \in (0, \bar{\delta}) \\ 1 & \text{if } \delta = \bar{\delta}, \end{cases} \tag{6}$$

where

$$h(\delta) = \frac{\delta}{\alpha(\bar{\delta} - \delta)} - \frac{\alpha(\bar{\delta} - \delta)}{\delta},$$

and $\alpha > 0$ is a parameter dependent on δ_1. The smoothness of f in the interval of interest follows from the smoothness of g, which in turn can be inferred from the smoothness of LF and h. The following claim, whose proof is reported in the Appendix, ensures that g satisfies all the required properties.

Claim 1. *Given the interval $[0, \overline{\delta}]$ and $\delta_1 \geq 0$, there exists an $\alpha > 0$ such that the function g, defined by equation* (6)*, with parameter α satisfies all the conditions* (1)–(5)*.*

In order to conclude the proof of the lemma, we need to make sure that $f(\delta)$ is a strict convex combination of x_0, x_1, x_2 or, equivalently, that it is contained in the interior of the triangle having vertices x_0, x_1, x_2. First, consider the line r_{01} passing through x_0 and x_1. For all $\delta \in [0, \bar{\delta}]$, the point $f(\delta)$ can be in one of three positions relative to r_{01}: it can be in the open half-plane delimited by r_{01} and containing x_2; it can be in the opposite open half-plane; or it can lie on r_{01} itself. By construction, $f(0) = x_0 \in r_{01}$. For $i = 0, 1$, decompose c_i as $c_i^{01} + c_i^{\perp 01}$, where c_i^{01} is the projection of c_i on the direction of r_{01} and $c_i^{\perp 01}$ is the projection on the direction that is orthogonal to r_{01} and lies in the plane identified by x_0, x_1, x_2. Clearly, it holds $c_0^{\perp 01} = 0$ (c_0 is parallel to r_{01}). Hence, only the vector $c_1^{\perp 01}$ is responsible for changing the position of $f(\delta)$ w.r.t. r_{01}. Since f starts on r_{01}, ends up in x_2, and uses only convex combinations of c_0 and c_1, it follows that f lies entirely in the half-plane delimited by r_{01} and containing x_2.

Next, consider the line r_{12} passing through x_1 and x_2. Similarly to the previous case, decompose c_0 and c_1 w.r.t. r_{12}. Now, we have $c_1^{\perp 12} = 0$ because c_1 is parallel to r_{12}, and only the vector $c_0^{\perp 12}$ is responsible for changing the position of $f(\delta)$ w.r.t. r_{12}. Since f starts in the semi-plane delimited by r_{12} and containing x_0 and ends up on r_{12}, it cannot reach the opposite semi-plane in the meanwhile, otherwise it would not be able to go back to r_{12}.

Finally, consider the line r_{02} passing through x_0 and x_2. In this case, both vectors $c_0^{\perp 02}$ and $c_1^{\perp 02}$ can modify the position of f w.r.t. r_{02}. Since f starts in $x_0 \in r_{02}$ and ends up in $x_2 \in r_{02}$, the (parallel) vectors $c_0^{\perp 02}$ and $c_1^{\perp 02}$ point in opposite directions. Then, notice that the function $g(\delta)$, which regulates the convex combination between c_0 and c_1, is strictly monotonic. Hence, as long as $(1 - g(\delta))c_0^{\perp 02}$ prevails over $g(\delta)c_1^{\perp 02}$, the trajectory moves away from r_{02} in the direction of x_1. After that, the trajectory goes back to r_{02} and eventually reaches it in x_2. Being $g(\delta)$ monotonic, it is not possible for f to go beyond r_{02} and then go back to x_2. ∎

Given a ($\mathcal{C}_\text{d}$- or $\mathcal{C}_\text{s}$-) witness $\xi = (f, \delta_f)$ and a convex polyhedron P, let Δ_ξ^P be the set of delays $\delta \leq \delta_f$ such that f lies in P in an open interval around δ. Formally,

$$\Delta_\xi^P = \big\{0 \leq \delta \leq \delta_f \,\big|\, \exists \gamma > 0 \, \forall \delta' \in (\delta - \gamma, \delta + \gamma) \cap [0, \delta_f] : f(\delta') \in P\big\}.$$

We say that ξ is *P-canonical* if either $\Delta_\xi^P = \emptyset$ or $f(\delta) \in P$ for all $\delta \in (\inf \Delta_\xi^P, \sup \Delta_\xi^P)$. The definition implies that once a P-canonical witness spends a positive amount of time in P and then exits from it, it can only return to P for instantaneous visits (i.e., in isolated time points). The following example justifies this definition, showing that witnesses may need to make multiple instantaneous visits to the same polyhedron.

Example 1. *Consider the scenario in Figure 6, where all polyhedra called A must be avoided and the flow constraint F is shown on the right-hand side. The A-polyhedra are topologically open and adjacent to P. Moreover, A_1 is weakly adjacent to A_3 and A_4 (i.e., their closures intersect), and so on for the other A-polyhedra.*

In order to go from x to G while avoiding A, a witness must cross the line P (which could be one of the patches of $\overline{A}$) multiple times. Notice that it is not possible for the differentiable witness to follow the line P and then reach G, because to do so it would either pass through some A-polyhedron or be non-differentiable when leaving P.

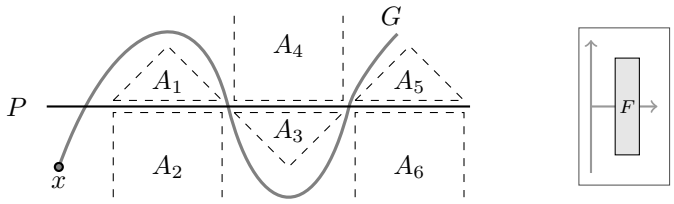


Figure 6: Witnesses may need to traverse the same convex polyhedron P multiple times. Polyhedra A_2, A_4, and A_6 are drawn truncated.

For a non-convex polyhedron B, we say that the witness ξ is B-canonical if it is P-canonical for all $P \in [\![B]\!]$. The following lemma ensures that smooth canonical witnesses are sufficient to determine which points belong to $RWA^{\mathcal{C}_\mathrm{d}}$.

Lemma 3 (Canonicity). *For all polyhedra H, trajectories f, and times $\delta_f \geq 0$, such that f lies in H in $[0, \delta_f]$, there exists an H-canonical smooth trajectory g and a time δ_g such that: $f(0) = g(0)$, $f(\delta_f) = g(\delta_g)$, $\dot{f}(0) = \dot{g}(0)$, and g lies in H until time δ_g.*

Proof. Let $\xi = (f, \delta_f)$ and $P \in [\![H]\!]$, we first prove that ξ can be modified to become P-canonical. If $\Delta_\xi^P = \emptyset$, ξ is already P-canonical. Otherwise, we prove the following claim.

Claim 2. *There exists a pair $\zeta = (g, \delta_g)$, where g is an admissible trajectory starting from $f(0)$ and $\delta_g \geq 0$, such that:*

(i) ζ is P-canonical;

(ii) $\Delta_\zeta^P \neq \emptyset$ and $\inf \Delta_\zeta^P = \inf \Delta_\xi^P$;

(iii) $g(\delta) = f(\delta)$, for all delays $\delta \in [0, \delta^P]$;

(iv) $g(\delta) = f(\delta + \sup \Delta_\xi^P - \sup \Delta_\zeta^P)$, for all delays $\delta \in [\sup \Delta_\zeta^P, \delta_g]$;

(v) g is smooth in the interval $(\inf \Delta_\zeta^P, \sup \Delta_\zeta^P)$, it holds $\dot{g}(\inf \Delta_\zeta^P) = \dot{f}(\inf \Delta_\zeta^P)$ and $\dot{g}(\sup \Delta_\zeta^P) = \dot{f}(\sup \Delta_\xi^P)$, and g has all the higher-order derivatives equal to 0 in $(\inf \Delta_\zeta^P)^+$ and $(\sup \Delta_\zeta^P)^-$.

Proof of Claim 2. Let $\gamma_0 = \inf \Delta_\xi^P$ and $\gamma_2 = \sup \Delta_\xi^P$. By definition, $\gamma_0 < \gamma_2$. Let γ_1 be a delay such that $\gamma_0 < \gamma_1 < \gamma_2$, with $f(\gamma_1) \in P$. For all $k = 0, \ldots, 2$, let $y_k = f(\gamma_k)$. The situation is depicted in Figure 7.

By definition of Δ_ξ^P, in all right-neighborhoods of γ_0 there is a time when f is in P. Hence, by applying Lemma 1 "backwards" from y_1 to y_0 we obtain a delay $\delta^* > 0$ such that $u \triangleq y_0 + \delta^* \dot{f}(\gamma_0) \in cl(P) \cap \{y_1\}\swarrow_{>0}$. Let t be an intermediate point on the line segment connecting u and y_1. Similarly, by applying Lemma 1 again from y_1 to y_2, we obtain a delay $\bar{\delta} > 0$ such that $u' \triangleq y_2 - \bar{\delta} \dot{f}(\gamma_2) \in cl(P) \cap \{y_1\} \nearrow_{>0}$. Let t' be an intermediate point on the line segment connecting u' and y_1.

We apply Lemma 2 three times: first, to points y_0, u, and t; then, to points t, y_1, and t', and, finally, to points t', u', and y_2. We thus obtain three admissible trajectories which can be differentiably connected at t' and t forming an admissible trajectory f' with derivative $\dot{f}'(0) = \dot{f}(\gamma_0)$ and $\dot{f}'(\gamma_2') = \dot{f}(\gamma_2)$, for some $\gamma_2' > 0$ with $f'(\gamma_2') = y_2$. By connecting f' with f at points y_0 and y_2, we finally obtain the pair $\zeta = (g, \delta_g)$, where

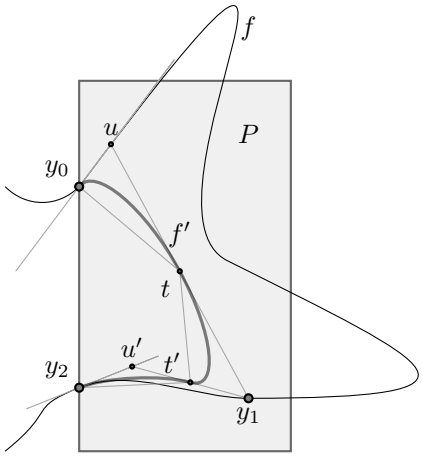


Figure 7: Witness trajectories do not need to spend a positive amount of time twice in the same convex polyhedron P (see Lemma 3).

g is a differentiable trajectory and $\delta_g = \delta_f + \gamma_2 - \gamma_2'$, satisfying $\inf \Delta_\zeta^P = \inf \Delta_\xi^P = \gamma_0$ and $\sup \Delta_\zeta^P = \gamma_2'$. Moreover, by Lemma 2, g is smooth in the interval (γ_0, γ_2') and all its higher-order derivatives at both γ_0^+ and $(\gamma_2')^-$ are equal to 0 *(end of claim proof)*.

The properties stated by the claim above imply that one can apply the construction twice, to two disjoint convex polyhedra P_1 and P_2, in order to obtain a trajectory that is $\{P_1, P_2\}$-canonical. Therefore, by repeatedly applying Claim 2 to all polyhedra in $[\![H]\!]$, we obtain an H-canonical pair $\eta = (g, \delta_g)$. Moreover, this construction ensures the stronger property that g is smooth in the whole interval $[0, \delta_g]$. Indeed, by the repeated construction above, η can spend a positive amount of time in each patch of H only once. So, let $P_1, \ldots, P_n$ be the sequence of patches in $[\![H]\!]$ visited, in that order, by g in the interval $[0, \delta_g]$ and such that $\Delta_\eta^{P_i} \neq \emptyset$ (i.e., for a positive amount of time). Let, in addition, $\gamma_0 = 0$, $\gamma_n = \delta_g$ and $\gamma_i \triangleq \sup \Delta_\eta^{P_i} = \inf \Delta_\eta^{P_{i+1}}$, for $i \in [1, n-1]$, be the time delay when g crosses from P_i to P_{i+1} (notice that $g(\gamma_i)$ may belong neither to P_i nor to P_{i+1}, but to some other patch which is visited instantaneously at that time instant). From condition *(v)* of Claim 2, it follows that g is smooth in the interval (γ_i, γ_{i+1}), for all $i \in [0, n-1]$. Moreover, all the higher-order derivatives of g at γ_i^- and at γ_i^+, for $i \in [1, n-1]$, are equal to 0. Since the left and right derivatives of any order coincide at every crossing points, g is infinitely differentiable in $[0, \delta_g]$ and the conclusion follows. ∎

By applying the above lemma to $\mathcal{C}_\mathrm{d}$-witnesses w.r.t. the polyhedron $\overline{A}$, we obtain the following.

Corollary 1. *For all disjoint polyhedra G and A, and for all points $x \in RWA^{\mathcal{C}_\mathrm{d}}(G, A)$, there exists a $\mathcal{C}_\mathrm{s}$–witness for x that is $\overline{A}$-canonical.*

The above corollary implies that the differentiable and the smooth versions of *RWA* coincide.

Theorem 3. *For all disjoint polyhedra G and A, it holds $RWA^{\mathcal{C}_\mathrm{s}}(G, A) = RWA^{\mathcal{C}_\mathrm{d}}(G, A)$.*

4. Computing Reach-While-Avoiding

Thanks to Theorem 3, we can restrict our attention to differentiable trajectories. Hence, we shall drop the corresponding superscript and write *Adm* for $Adm^{\mathcal{C}_\mathrm{d}}$ and *RWA* for $RWA^{\mathcal{C}_\mathrm{d}}$. In this section we show how the results obtained so far can be used to define an exact symbolic algorithm for computing $RWA(G, A)$. In particular, we proceed by collecting $\mathcal{C}_\mathrm{d}$-witnesses (in the following, simply "witnesses") into classes, called

types, based on the sequence of polyhedra that they traverse before reaching the target region. We then show how to compute the set of points that have a canonical witness of a specific type. Since the possible types of canonical witnesses form a finite set, we obtain an algorithm for *RWA*.

A *type* is a finite sequence of (possibly non-convex) polyhedra, each annotated with a 0 superscript or a $>$ superscript, in an alternating fashion. For example, if P, Q, R are polyhedra, $P^0Q^>R^0R^>$ and $P^>Q^0R^>$ are legitimate types, whereas $P^0Q^>R^>$ and $P^0Q^0R^>$ are not. We shall refer to types beginning with a P^0 set as *0-types* and to the ones beginning with a $P^>$ set as *non-0-types*.

We use types to represent the sequence of convex polyhedra traversed by a trajectory. Given a type $\mathbf{T} = P_0^{\diamond_0} P_1^{\diamond_1} \dots P_l^{\diamond_l}$ ($\diamond_i \in \{0, >\}$), we say that a $\mathcal{C}_\mathrm{d}$-trajectory (in the following, simply "trajectory") f has type $\mathbf{T}$ if there exists a time $\delta_f \geq 0$ and a partition of the interval $[0, \delta_f]$ in a finite sequence of intervals $I_0, I_1, \dots, I_l$ such that f lies in P_i during each interval I_i and, moreover, I_i is a singular interval iff $\diamond_i = 0$.

For example, a trajectory f of type $P^0Q^>R^0R^>$ starts in P, immediately enters Q and spends some time in it, then reaches R and stays in it for a while. A trajectory g of type $P^0Q^>Q^0R^>$ is similar to f, except for the fact that f has a first time instant in which it lies in R, whereas g has a last time instant in which it lies in Q before entering R. Let $Ext(\mathbf{T})$ (for *extension*) denote the set of points from which at least one trajectory of type $\mathbf{T}$ starts. We denote the empty type by ε and we set $Ext(\varepsilon) = \mathbb{R}^n$ (the whole ambient space). A *convex type* is a type $\mathbf{T}$ that only contains convex polyhedra.

As before, we fix the target convex polyhedron G and the avoidance polyhedron A, and we assume w.l.o.g. that G is one of the patches of $\overline{A}$. Recall that a witness for a point x is a pair $\xi = (f, \delta_f)$, such that f is an admissible trajectory that starts from x, reaches G at time δ_f, and avoids A at all intermediate times. A $\overline{A}$-canonical witness does not return to a patch of $\overline{A}$ once it has spent a positive amount of time there.

We denote by *CTypes* (for *canonical types*) the set of all types containing only convex polyhedra in $[\![\overline{A}]\!]$ and such that: no polyhedron P appears twice as $P^>$ and the last polyhedron in the sequence is G. Formally, *CTypes* is the smallest set of types satisfying the following:

- $\{G^0, G^>\} \subseteq$ *CTypes*;
- if $\mathbf{T} \in$ *CTypes* is a non-0-type and $P \in [\![\overline{A}]\!]$, then $P^0\mathbf{T} \in$ *CTypes*;
- if $\mathbf{T} \in$ *CTypes* is a 0-type, $P \in [\![\overline{A}]\!]$ and $P^>$ does not occur in $\mathbf{T}$, then $P^>\mathbf{T} \in$ *CTypes*.

Notice that the length of the types in *CTypes* is bounded by $2k + 1$, where k is the number of patches in $\overline{A}$. Therefore, *CTypes* is a finite set. In addition, every canonical type is also a convex type.

It is easy to verify that all $\overline{A}$-canonical witnesses have a type in *CTypes*, whereas not all types in *CTypes* correspond to actual witnesses. The following result formalizes the relation between $RWA(G, A)$ and *CTypes*.

Theorem 4. *It holds that*

$$RWA^{\mathcal{C}_\mathrm{d}}(G, A) = \bigcup_{\mathbf{T} \in CTypes} Ext(\mathbf{T}).$$

Proof. The direction $\bigcup_{\mathbf{T} \in CTypes} Ext(\mathbf{T}) \subseteq RWA(G, A)$ follows immediately from the definitions of the operators involved. For the other direction, assume $x \in RWA(G, A)$. Then there exists a trajectory $f \in Adm(x)$ reaching G while always avoiding A. By Lemma 3 there is a witness ξ from x which is $\overline{A}$-canonical and ends in G. Since ξ is $\overline{A}$-canonical, there is a type $\mathbf{T} \in$ *CTypes* such that ξ is of type $\mathbf{T}$, and therefore $x \in Ext(\mathbf{T})$. The conclusion follows. ∎

As a consequence of the above theorem, in order to compute *RWA* we only need to be able to compute extensions of convex types.

5. Computing *Ext* of a Convex Type

In this section, we show how to reduce $Ext(\mathbf{T})$ to a sequence of steps involving at most 3 polyhedra at a time. To this aim, define the following abbreviations:

$$Ext_2(P, Q) = Ext(P^0Q^>), \qquad Ext_3(P, Q, R) = Ext(P^>Q^0R^>).$$

We start with two technical lemmas. The first one shows that Ext_3 distributes over unions of polyhedra in its third argument.

Lemma 4. *For all convex polyhedra P, Q, and polyhedra B, we have $Ext_3(P,Q,B) = \bigcup_{R\in[\![B]\!]} Ext_3(P,Q,R)$ and $Ext_2(P,B) = \bigcup_{R\in[\![B]\!]} Ext_2(P,R)$.*

Proof. We prove the thesis for Ext_3, as the one for Ext_2 is analogous. The $\supseteq$ inclusion being obvious, let us focus on the other direction. Let $x \in Ext_3(P,Q,B)$ and let f be a $\mathcal{C}_{\mathrm{d}}$-trajectory of type $P^{>}Q^0B^{>}$. Let $\delta_1 > 0$ be the time instant when f reaches Q. Define g as the suffix of f starting from time δ_1, i.e., $g(\delta) = f(\delta+\delta_1)$, for all $\delta \geq 0$. Notice that g is a trajectory of type $Q^0B^{>}$. By Lemma 3, there exists a $\mathcal{C}_{\mathrm{s}}$-trajectory g' that is B-canonical. Moreover, g and g' have the same value and first derivative at 0, which are equal to $f(\delta_1)$ and $\dot{f}(\delta_1)$, respectively. Hence, we can append g' to the prefix of f up to time δ_1, thus obtaining a differentiable B-canonical trajectory f' of type $P^{>}Q^0B^{>}$. By B-canonicity, there exists a first patch $R \in [\![B]\!]$ in which f' spends a positive amount of time. As a consequence, f' is also of type $P^{>}Q^0R^{>}$, as required. ∎

The following technical lemma, proved in the Appendix, states that each trajectory can be modified in such a way that it starts with a straight segment of positive duration, while retaining its type. This is needed in Lemma 6 to perform the composition of trajectories depicted in Figure 8.

Lemma 5. *For all types $\mathbf{T}$ and $x \in Ext(\mathbf{T})$ there exist a trajectory $f \in Adm(x)$ of type $\mathbf{T}$, a positive delay δ^*, and a slope $c \in F$ such that $\dot{f}(\delta) = c$ for all $\delta \in [0,\delta^*)$.*

We now show that the extension of a type $\mathbf{T}$ remains unchanged if a proper suffix of $\mathbf{T}$ is replaced with its extension. This property provides the inductive step for the computation of $Ext(\mathbf{T})$, for all types $\mathbf{T}$. In particular, there is a way to partition any type $\mathbf{T}$ into a sequence of two types $\mathbf{T}'\mathbf{T}''$, so that the extension of $\mathbf{T}$ coincides with the extension of type $\mathbf{T}'Ext(\mathbf{T}'')^{>}$. The crucial observation is that Lemma 2 allows us to connect in a differentiable way a trajectory of type $\mathbf{T}'Ext(\mathbf{T}'')^{>}$ with a trajectory of type $\mathbf{T}''$, thus providing a trajectory of type $\mathbf{T}$.

Given a set $P \subseteq \mathbb{R}^n$, we say that a trajectory f *lingers* in P if there exists $\delta > 0$ such that $f(\delta') \in P$, for all $\delta' \in (0,\delta)$.

Lemma 6. *Assume that $Ext(\mathbf{T})$ is a polyhedron for every convex type $\mathbf{T}$. Then, for all non-empty convex types $\mathbf{T}$ the following holds:*

$$Ext(\mathbf{T}) = \begin{cases} P & \text{if } \mathbf{T} = P^0, \\ P \cap P{\swarrow_{>0}} & \text{if } \mathbf{T} = P^{>}, \\ Ext_3(P,Q,Ext(\mathbf{T}')) & \text{if } \mathbf{T} = P^{>}Q^0\mathbf{T}', \\ Ext_2(P,Ext(Q^{>}\mathbf{T}')) & \text{if } \mathbf{T} = P^0Q^{>}\mathbf{T}'. \end{cases} \tag{7}$$

Proof. The first case is immediate, as for any $x \in P$, any $f \in Adm(x)$ is a trajectory of type P^0. For the second case, the left-hand side asks for an admissible trajectory f lying within P for a positive amount of time. This corresponds to the definition of the convex polyhedron $P \cap P{\swarrow_{>0}}$, hence the conclusion.

Third case. Assume that $\mathbf{T} = P^{>}Q^0\mathbf{T}'$ and let $x \in Ext(\mathbf{T})$. Then there is a trajectory f, which is of type $\mathbf{T}$ in some interval $[0,\delta_f]$, such that $x = f(0) \in P$ and either $\mathbf{T}'$ is empty or $\mathbf{T}'$ has the form $R^{>}\mathbf{T}''$, for some convex polyhedron R and type $\mathbf{T}''$. In the former case, it holds that $Ext(\mathbf{T}') = \mathbb{R}^n$, and we immediately obtain $x \in Ext_3(P,Q,Ext(\mathbf{T}'))$. In the latter case, for some $0 < \delta_1 < \delta_2 \leq \delta_f$, it holds that $f(\delta') \in P$ for all $\delta' \in [0,\delta_1)$, $f(\delta_1) \in Q$ and $f(\delta') \in R$ for all $\delta' \in (\delta_1,\delta_2]$. Since, however, $\mathbf{T} = P^{>}Q^0R^{>}\mathbf{T}''$, for each $\delta' \in (\delta_1,\delta_2)$, $f(\delta') \in Ext(R^{>}\mathbf{T}'') \subseteq R$. Hence we obtain that $x \in Ext_3(P,Q,Ext(\mathbf{T}'))$.

For the other direction, suppose $x \in Ext_3(P,Q,Ext(\mathbf{T}'))$. Let us consider the case $\mathbf{T}' = R^{>}\mathbf{T}''$ (see Figure 8), since the case $\mathbf{T}' = \epsilon$ is trivial. Notice that $Ext(\mathbf{T}') \subseteq R$. By assumption, $Ext(\mathbf{T}')$ is a polyhedron, and by Lemma 4 there is a patch E of $Ext(\mathbf{T}')$ such that $x \in Ext_3(P,Q,E)$. Let f be a trajectory of type

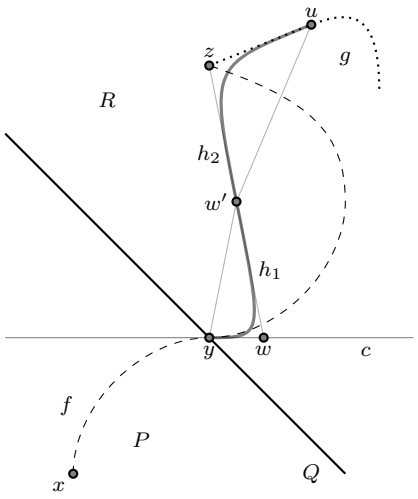


Figure 8: Connecting two witnesses (the dashed line f starting from x and the dotted line g starting from z) into a single one (Lemma 6).

$P^{>}Q^{0}E^{>}$ in some interval $[0, \delta_f]$. Moreover, let δ_1, δ_2 be two delays such that $f(\delta) \in P$ for all $\delta \in [0, \delta_1)$, $f(\delta_1) \in Q$, and $f(\delta) \in E$ for all $\delta \in (\delta_1, \delta_2]$. Let us set $y = f(\delta_1)$, $z = f(\delta_2)$, and $c = \dot{f}(\delta_1)$. Since $z \in E \subseteq \mathit{Ext}(\mathbf{T}')$, by Lemma 5 there exists a trajectory g from z which: *(i)* is of type $\mathbf{T}'$, *(ii)* starts with a straight line segment. Since f is of type $\mathbf{T}'$, it also lingers in R. Let $\delta^* > 0$ be such that $u = g(\delta^*)$ still lies within R and along the initial straight line segment of g with slope c'.

By applying Lemma 1 backwards from z (i.e., taking z as the point x of the statement of the lemma and y as the point $f(\hat{\delta})$) and the convex polyhedron R, we obtain a delay $\hat{\delta} > 0$ such that the point $w \triangleq y + \hat{\delta}c$ belongs to $cl(R)$ and can reach z along an admissible straight trajectory of positive duration. In addition, let w' be the midpoint between $w \in cl(R)$ and $z \in R$, hence $w' \in R$ as well. We can now apply Lemma 2 twice: first, to the points y, w and w'; then, to the points w', z, u. We thus obtain two admissible trajectories, h_1 and h_2, both contained in R and that can be differentiably connected in w' (having the same first derivative at w'). Their concatenation gives us a trajectory h having first derivative equal to c in y and c' in u. Finally, by connecting f with h at y and h with g at u we obtain a trajectory of type $P^{>}Q^{0}\mathbf{T}'$ starting from x. We can then conclude that $\mathit{Ext}(P^{>}Q^{0}\mathbf{T}') = \mathit{Ext}_3(P, Q, \mathit{Ext}(\mathbf{T}'))$.

Fourth case. Assume that $\mathbf{T} = P^{0}Q^{>}\mathbf{T}'$ and let $x \in \mathit{Ext}(\mathbf{T})$. Then there is a trajectory f of type $\mathbf{T}$ starting from x. By construction, there exists $\delta_1 > 0$ such that $f(\delta) \in Q$, for all $\delta \in (0, \delta_1]$. Let $\delta \in (0, \delta_1]$ and $y = f(\delta)$. The suffix of f starting from time δ proves that y belongs to $\mathit{Ext}(Q^{>}\mathbf{T}')$. As a consequence, f starts in P, immediately enters $\mathit{Ext}(Q^{>}\mathbf{T}')$, and lingers in it for a non-empty time interval. Therefore, we obtain that $x \in \mathit{Ext}_2(P, \mathit{Ext}(Q^{>}\mathbf{T}'))$.

For the other direction, notice first that the assumption ensures that $\mathit{Ext}(Q^{>}\mathbf{T}')$ is a polyhedron. Let $x \in \mathit{Ext}_2(P, \mathit{Ext}(Q^{>}\mathbf{T}'))$, there exists a trajectory f starting from $x \in P$ that lingers in $\mathit{Ext}(Q^{>}\mathbf{T}')$ for a non-empty time interval $(0, \delta)$, i.e. $f(\delta') \in \mathit{Ext}(Q^{>}\mathbf{T}')$ for all $\delta' \in (0, \delta)$. By considering again Figure 8 and taking $x = y$ in the figure, we can apply the same construction as in the previous case and obtain a trajectory starting from x and of type $P^{0}Q^{>}\mathbf{T}'$. Hence, we conclude that $x \in \mathit{Ext}(P^{0}Q^{>}\mathbf{T}')$. As a consequence, $\mathit{Ext}(P^{0}Q^{>}\mathbf{T}') = \mathit{Ext}_2(P, \mathit{Ext}(Q^{>}\mathbf{T}'))$. ∎

Assuming we can compute Ext_2 and Ext_3, and that their results are polyhedra, this section provides all the tools to compute $\mathit{RWA}(G, A)$. By Theorem 4, it is sufficient to compute $\mathit{Ext}(\mathbf{T})$ for all types $\mathbf{T} \in \mathit{CTypes}$. Each such $\mathbf{T}$ is a finite sequence of patches of $\overline{A}$, with the last patch being G (annotated as G^0 or $G^>$). Then, we iteratively compute $\mathit{Ext}(\mathbf{T})$, starting from the last patch and going backwards, according to the prescriptions of Lemma 6. Specifically, the algorithm proceeds by computing the extensions of non-zero

suffixes of $\mathbf{T}$. Assume that we have computed the extension of a proper non-zero suffix $\mathbf{T}'$ of $\mathbf{T}$. If there are at least two more patches in $\mathbf{T}$ (i.e., $\mathbf{T} = \dots P^{>}Q^{0}\mathbf{T}'$), then

$$Ext(P^{>}Q^{0}\mathbf{T}') = Ext_{3}(P, Q, Ext(\mathbf{T}')).$$

Otherwise, $\mathbf{T} = P^{0}\mathbf{T}'$. In this case,

$$Ext(P^{0}\mathbf{T}') = Ext_{2}(P, Ext(\mathbf{T}')).$$

Depending on how type $\mathbf{T}$ ends, the algorithm starts with one of the following:

$$Ext(G^{>}) = G \cap G_{\swarrow > 0}, \qquad Ext(P^{>}G^{0}) = Ext_{3}(P, G, \mathbb{R}^{n}),$$

and, in case $\mathbf{T} = G^{0}$, we simply have $Ext(G^{0}) = G$. The following two sections deal with the computation of the operators Ext_{2} and Ext_{3} applied to polyhedra.

5.1. Computing Ext_{2} on Polyhedra

We now show how to reformulate the computation of the operator Ext_{2} in terms of straight trajectories only. The main results of this section and the next one are summarized in Figure 11.

We recall the following result from the literature, which guarantees that every point reachable from a point x along an admissible and differentiable trajectory can also be reached from x along an admissible straight trajectory.

Proposition 1. [1] *For all points $x \in \mathbb{R}^{n}$, if there is a trajectory $f \in Adm(x)$ and a time $\delta > 0$ such that $f(\delta) = y$, then there is a slope $c \in F$ such that $y = x + \delta c$.*

Notice that the proof of Proposition 1 only requires that f be continuous, therefore the result holds of all the classes of trajectories considered in the present paper.

The following result shows that Ext_{2} can be expressed in terms of straight directions and then easily computed with basic polyhedral operations.

Theorem 5. *For all disjoint convex polyhedra P and Q, it holds $Ext_{2}(P, Q) = P \cap cl(Q) \cap Q_{\swarrow}$. In particular, $Ext_{2}(P, Q)$ is a convex polyhedron.*

Proof. First, notice that, by definition, we have the following:

$$Ext(P^{0}Q^{>}) = \left\{ x \in P \,\middle|\, \exists f \in Adm(x), \delta > 0 \, \forall \delta' \in (0, \delta] : f(\delta') \in Q \right\}. \tag{8}$$

Now, we can prove the two sides of the equivalence.
($\subseteq$) Let $x \in Ext(P^{0}Q^{>})$ and let $f \in Adm(x)$ be the trajectory, whose existence is postulated in (8). Since f lingers in Q, in each neighborhood of x there is a point in Q. Hence, $x \in cl(Q)$. Moreover, Proposition 1 implies that $x \in Q_{\swarrow}$.

($\supseteq$) Let $x \in P \cap cl(Q) \cap Q_{\swarrow}$ and let $y \in Q$ such that $y = x + \bar{\delta}c$, for suitable $\bar{\delta} \geq 0$ and $c \in F$. Since P and Q are disjoint, it holds $\bar{\delta} > 0$. Then, the trajectory $f(\delta) = x + \delta c$ lingers in Q and proves that $x \in Ext(P^{0}Q^{>})$. ∎

5.2. Computing Ext_{3} on Polyhedra

In words, $Ext_{3}(P, \hat{P}, P')$ contains the points of P that can reach and spend a positive amount of time in P', via a trajectory that remains within $P \cup P'$ at all times, except for an intermediate time instant δ_{1} in which the trajectory is in $\hat{P}$. Notice that the conditions above imply that $f(\delta_{1}) \in cl(P) \cap cl(P')$. Hence,

$$Ext_{3}(P, \hat{P}, P') = Ext_{3}(P, \hat{P} \cap cl(P) \cap cl(P'), P'),$$

and we can assume w.l.o.g. that $\hat{P}$ is included in $cl(P) \cap cl(P')$.

5.2.1. From Trajectories to Straight Trajectories: First Attempt

In order to compute $Ext_\mathcal{3}$, we try to reformulate it in terms of straight trajectories. If we simply replace the arbitrary trajectory $f \in Adm(x)$ in the definition of $Ext_\mathcal{3}$ with a straight trajectory of slope c, we obtain the following operator:

$$SExt_\mathcal{3}(P, \hat{P}, P') = \big\{x \in P \,\big|\, \exists c \in F, \delta_1 > 0, \delta_2 > \delta_1 : \\ \forall \delta \in (0, \delta_1) : x + \delta c \in P \text{ and } x + \delta_1 c \in \hat{P} \text{ and } \forall \delta \in (\delta_1, \delta_2] : x + \delta c \in P'\big\}.$$

Intuitively, these are the points of P that can reach a point in P' following a straight direction while remaining in $P \cup P'$ at all times, except for a single point in $\hat{P}$. Clearly, $SExt_\mathcal{3}(P, \hat{P}, P') \subseteq Ext_\mathcal{3}(P, \hat{P}, P')$. In addition, any point in P that can reach $SExt_\mathcal{3}(P, \hat{P}, P')$ also belongs to $Ext_\mathcal{3}(P, \hat{P}, P')$:

Lemma 7. *For all convex polyhedra P, $\hat{P}$ and P' the following holds:*

$$P \cap SExt_\mathcal{3}(P, \hat{P}, P')_{\swarrow} \subseteq Ext_\mathcal{3}(P, \hat{P}, P').$$

Proof. The proof is illustrated in Figure 9. Let $x \in P \cap SExt_\mathcal{3}(P, \hat{P}, P')_{\swarrow}$ and let $y \in SExt_\mathcal{3}(P, \hat{P}, P')$ be such that $x \in \{y\}_{\swarrow}$. There exist $c \in F$ and $0 < \delta_1 < \delta_2$ satisfying the definition of $SExt_\mathcal{3}(P, \hat{P}, P')$ for y. Let $z = y + \delta_1 c$, by construction it holds $z \in \hat{P} \cap cl(P) \cap cl(P')$. By applying Lemma 2 with $x_0 = x$, $x_1 = y$, and $x_2 = z$, we get a differentiable trajectory f in $Adm(x)$ from x to z whose derivative in z is c. Moreover, f is contained in P, with the possible exception of $z \in \hat{P}$. Therefore, the concatenation of f and the straight trajectory $g(\delta) = z + \delta c$ is everywhere differentiable and crosses into P'. As a consequence, $x \in Ext_\mathcal{3}(P, \hat{P}, P')$. ∎

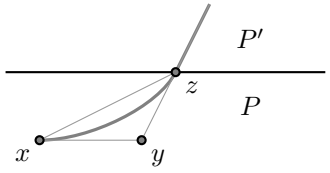


Figure 9: When a point $x \in P$ can reach another point y which is in $SExt_\mathcal{3}(P, \hat{P}, P')$, x can differentiably cross into P' (see Lemma 7). Here, $\hat{P} = \{z\}$.

Lemma 7, albeit providing a sound approximation of $Ext_\mathcal{3}$ in terms of straight trajectories, does not, unfortunately, ensure completeness. Indeed, there may be points that belong to $Ext_\mathcal{3}(P, \hat{P}, P')$ but not to $SExt_\mathcal{3}(P, \hat{P}, P')_{\swarrow}$.

Example 2. *Consider the scenario depicted in Figure 10, where P and P' are open convex polyhedra, $\hat{P}$ contains a single point z (the upper right corner of the closure of P), and the line segment A (which does not include $\hat{P}$) is a region to avoid. Given the flow constraint depicted on the right-hand side of the figure, it holds $SExt_\mathcal{3}(P, \hat{P}, P') = \emptyset$, as no straight trajectory leads from P to P' passing through $\hat{P}$.*

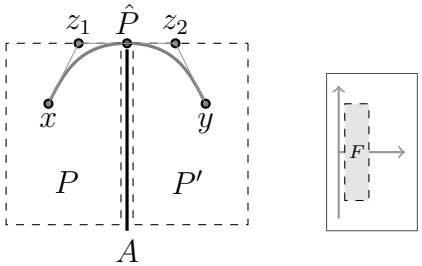


Figure 10: Reaching points in $SExt_\mathcal{3}$ is not necessary to be in $Ext_\mathcal{3}$.

Notice, however, that the point z_1, lying in the closure of P, is reachable from x by following a straight trajectory which always remains in the closure of P. Then, a straight trajectory, with derivative c, leads from z_1 to a point z_2, which lies in the closure of P', without ever leaving the closures of the two polyhedra. Finally, z_2 can reach y in P', following a straight trajectory that never leaves the closure of P'. Therefore,

Lemma 2 applied with $x_0 = x$, $x_1 = z_1$ and $x_2 = z$ gives a differentiable trajectory from x to z, which never leaves P except for the end point $z \in \hat{P}$ and whose derivative in z is c (the straight direction from z_1 to z_2). Similarly, another application of Lemma 2, this time to $x_0 = z$, $x_1 = z_2$ and $x_2 = y$, gives a differentiable trajectory from z to y, which never leaves P' except for the starting point $z \in \hat{P}$ and whose derivative in z is, again, c. The concatenation of these two trajectories in z is depicted in Figure 10 and is a differentiable trajectory from x to y which never leaves $P \cup P'$ except for the single point $z \in \hat{P}$. Hence, we have that $x \in Ext_3(P, \hat{P}, P')$.

5.2.2. From Trajectories to Straight Trajectories: Second Attempt

The previous example suggests that the straight trajectory reformulation of Lemma 7, though not complete, can be extended, exploiting Lemma 2, by also allowing certain straight trajectories lying in the closures of P and P'. We therefore obtain the following operator:

$$\begin{aligned} CExt_3(P, \hat{P}, P') = \big\{ x \in cl(P) \cap P\nearrow \;\big|\; & \exists c \in F, 0 < \delta_1 < \delta_2 : \\ & \forall \delta \in (0, \delta_1) : x + \delta c \in cl(P) \text{ and } x + \delta_1 c \in \hat{P} \text{ and} \\ & \forall \delta \in (\delta_1, \delta_2) : x + \delta c \in cl(P') \text{ and } x + \delta_2 c \in cl(P') \cap P'\swarrow \big\}. \end{aligned}$$

We can simplify the above definition and remove the universal quantifications. Indeed, if $x \in cl(P)$ and $x + \delta_1\, c \in \hat{P} \subseteq cl(P)$, by convexity $x + \delta c$ is also in $cl(P)$, for all intermediate times δ. A similar argument holds for membership in $cl(P')$. As a consequence, by letting $Cross(Q, R)$ denote the set of points that can reach Q and then R along the same straight direction, i.e.,

$$Cross(Q, R) = \big\{ x \;\big|\; \exists c \in F, 0 < \delta_1 < \delta_2 : x + \delta_1 c \in Q \text{ and } x + \delta_2 c \in R \big\},$$

we obtain the following equation:

$$CExt_3(P, \hat{P}, P') = cl(P) \cap P\nearrow \cap\ Cross\big(\hat{P}, cl(P') \cap P'\swarrow\big). \tag{9}$$

We can prove that $CExt_3$ applied to convex polyhedra is a convex set.

Lemma 8. *For all convex polyhedra P, $\hat{P}$ and P', $CExt_3(P, \hat{P}, P')$ is a convex set.*

Proof. Let $x, x' \in CExt_3(P, \hat{P}, P')$. By definition, there exist directions $c, c' \in F$, and positive reals $\delta_1, \delta_2, \delta_1', \delta_2'$, such that $y \triangleq x + \delta_1 c \in \hat{P}$, $x + \delta_2 c \in cl(P') \cap P'\swarrow$, $y' \triangleq x' + \delta_1' c' \in \hat{P}$, and $x' + \delta_2' c' \in cl(P') \cap P'\swarrow$.

Let $x'' = a\, x + (1 - a)\, x'$, for some $a \in (0, 1)$; we identify a convex combination of c and c' that, starting from x'', reaches $\hat{P}$ and then spends a positive amount of time in $cl(P') \cap P'\swarrow$. To this aim, we set $c'' = bc + (1 - b)c'$. To find the unknown b, we require that the the half-line starting from x'' with direction c'' reaches the appropriate convex combination of y and y': $x'' + \delta_1'' c'' = a\, y + (1 - a)\, y'$. A solution to the above equation is the following:

$$b = \frac{\delta_1\, a}{\delta_1''}, \qquad \text{where } \delta_1'' = a\delta_1 + (1 - a)\delta_1'.$$

The point $a\, y + (1 - a)\, y'$ belongs to $\hat{P}$ because it is a convex combination of two points in $\hat{P}$, and by convexity of $\hat{P}$. A similar argument shows that there is a time $\delta_2'' > \delta_2'$ such that the points of the type $x'' + \gamma c''$ belong to $cl(P') \cap P'\swarrow$, for all $\gamma \in (\delta_1'', \delta_2'')$. This proves that $x'' \in CExt_3(P, \hat{P}, P')$, and our thesis. ∎

The following theorem, whose proof can be found in the Appendix, shows that $CExt_3$ enables a sound and complete reformulation of Ext_3 in terms of straight trajectories.

Theorem 6. *For all convex polyhedra P, $\hat{P}$ and P' the following holds:*

$$Ext_3(P, \hat{P}, P') = P \cap CExt_3(P, \hat{P}, P')\swarrow\ .$$

In conclusion, Theorems 5 and 6 allow us to compute $Ext_3(P, \hat{P}, P')$ provided we can compute $CExt_3(P, \hat{P}, P')$, and in particular $Cross(\hat{P}, cl(P') \cap P'\swarrow)$. Indeed, that is the subject of the following two sections.

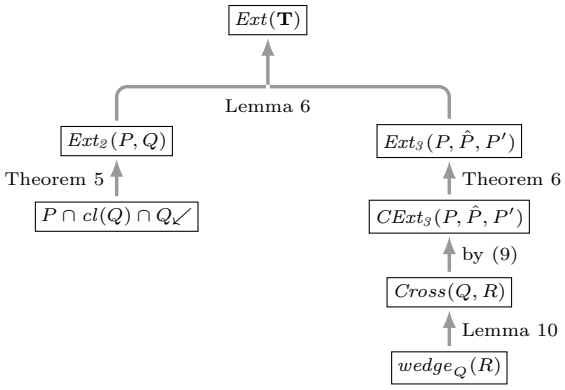


Figure 11: The main steps required for computing the extension of a convex type.

5.2.3. Geometric Primitives

An *affine combination* of two points x and y is any point $z = ax + (1 - a)y$, for $a \in \mathbb{R}$. An *affine set* is any set of points closed under affine combinations. The empty set, a single point, a line, a hyperplane, and the whole ambient space are all examples of affine sets. Given a convex polyhedron P, the *affine hull* of P, denoted $ahull(P)$, is the smallest affine set containing P.

Given a polyhedron $P \subseteq \mathbb{R}^n$, the *affine dimension* of P is the natural number $k \leq n$, such that the maximum number of affinely independent points in P is $k+1$. The *relative interior* of a convex polyhedron P, denoted $rint(P)$, is the interior of P, relative to its affine hull. More formally, $x \in rint(P)$ if and only if there is a ball $N_\epsilon(x)$ of radius $\epsilon > 0$ centered in x, such that $N_\epsilon(x) \cap ahull(P) \subseteq P$. Similarly, given a polyhedron P, we call the set N a *relative neighborhood* of a point $x \in P$, if there is a ball $N_\epsilon(x)$ of radius $\epsilon > 0$, such that $N = N_\epsilon(x) \cap ahull(P)$. Intuitively, a relative neighborhood of x is a neighborhood of x relative to the the affine hull of P. Finally, we say that P is *relatively open* if it is open relative to its affine hull.

The *point reflection* of a point y w.r.t. a point x, in symbols $mirror_x(y)$, is the point y', beyond x along the line connecting x and y, such that $\|y - x\| = \|y' - x\|$. Similarly, one can define the reflection $mirror_Q(y)$ of y w.r.t. an affine set Q, as the point reflection of y w.r.t. the *orthogonal projection* of y on Q. The above definitions can be extended to sets of points P, giving rise to the reflections $mirror_x(P)$ and $mirror_Q(P)$. For a convex polyhedron Q which is not necessarily an affine set, we will abuse the notation and write $mirror_Q(P)$ when we mean $mirror_{ahull(Q)}(P)$.

The affine hull and the relative interior can be easily computed using the standard "double description" of convex polyhedra via constraints and generators [14]. Moreover, it is well known that the reflection of a point w.r.t. a given affine set is a linear transformation, and as such it is exactly computable starting from a representation of the affine set. The details are beyond the scope of the present paper.

5.2.4. Computing Cross

The main difficulty we have to face in order to compute $Cross(Q, R)$ is to ensure that the points collected in the set can reach Q and then R following a *single* admissible straight direction. Assume first that Q contains a single point x; this case is illustrated in Figure 12. In order to compute the set of points that can reach a given polyhedron R along a single admissible direction that passes through x, we can proceed as follows. We first compute the set W of points reachable from x following some (non-necessarily admissible) direction that passes through R. Then we compute the mirror image of W w.r.t. x (patterned area in the figure), to obtain the set of points that can reach R via x following a single straight direction. Finally, we intersect the resulting set with the positive pre-flow of x, thus obtaining the set of points that can reach R via x following a single straight and admissible direction. In order to generalize the procedure to the case where the single point x is replaced by an arbitrary convex polyhedron Q, we first introduce the operator

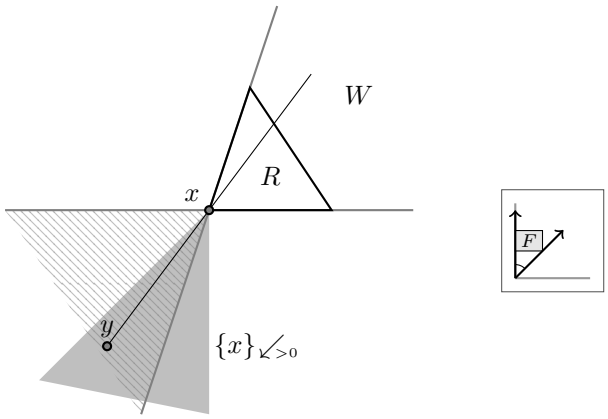


Figure 12: The patterned area is $mirror_x(W)$ and the intersection of the patterned and shaded areas is $Cross(\{x\}, R)$.

$wedge_Q(R)$, which collects the points reachable from some point x in Q following some (non-necessarily admissible) direction that passes through R. Formally, given two convex polyhedra Q and R, let

$$wedge_Q(R) = \left\{x + \delta\, c \,\middle|\, x \in Q, \delta \geq 0 \text{ and } c \in R \oplus \{-x\}\right\}$$

In other words, the direction c can be any vector starting from x and leading to a point in R. The procedure described above for the case $Q = \{x\}$ leads to the following result:

$$Cross(x, R) = \{x\}_{\swarrow>0} \cap mirror_x(wedge_x(R)). \tag{10}$$

Unfortunately, if we replace the single point x with an arbitrary convex polyhedron Q, the above equivalence does not hold any longer, as the following example shows.

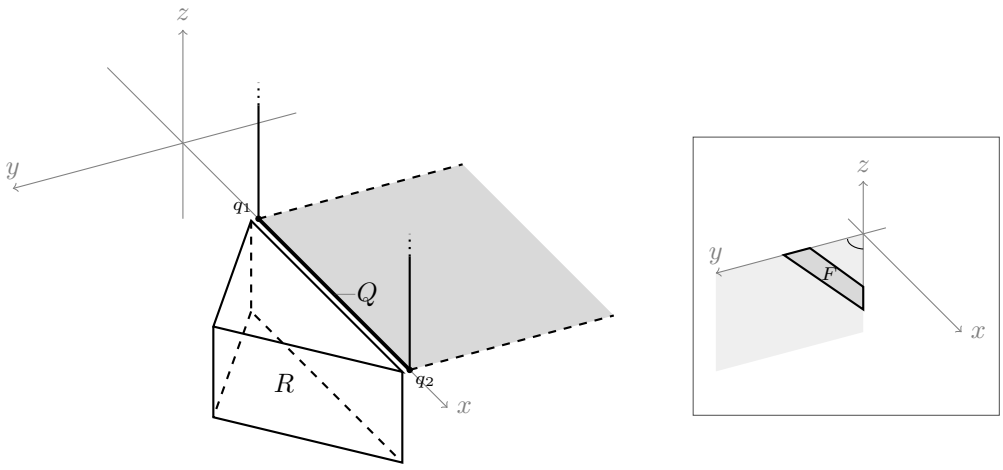


Figure 13: Example 3.

Example 3. *Consider the two convex polyhedra Q and R depicted in Figure 13, where Q is included in the closure of R and $q_1 = (1, 0, 0)$ and $q_2 = (3, 0, 0)$. Assume that the flow constraint allows the system to move on planes parallel to the yz-plane, as illustrated on the right-hand side of the picture. If we apply the procedure outlined above to Q and R we obtain: $W \triangleq wedge_Q(R) = \{(x, y, z) \mid y \geq 0$ and $z \leq 0\}$; $Z \triangleq mirror_Q(W) =$*

$\{(x,y,z) \mid y \leq 0 \text{ and } z \geq 0\}$; *finally,* $Q\swarrow_{>0} \cap Z = \{(x,y,z) \mid 1 \leq x \leq 3,\ y \leq 0, z \geq 0,\ \text{and } y+z > 0\}$, *where the additional constraint* $y + z > 0$ *removes* Q *from the result, since* Q *is disjoint from* $Q\swarrow_{>0}$ *in this case. The result does collect all the points in* $Cross(Q,R)$ *but also additional points that cannot cross into* R *via* Q*. Indeed, according to the definition,* $Cross(Q,R)$ *contains all the points in* $Q\swarrow_{>0} \cap Z$ *except for the two dashed half-lines lying on the* xy*-plane and passing through* q_1 *and* q_2*. In fact, no point on those half-lines can cross into* R *passing through* Q *following an admissible direction.*

Notice that, simply excluding the extreme points q_1 *and* q_2 *from* Q, i.e., *taking* $Q' = rint(Q)$ *instead of* Q, *would restore soundness of the result, but lose completeness. In the example, we would obtain the following*

$$Q'\swarrow_{>0} \cap\ mirror_{Q'}(wedge_{Q'}(R)) = \{(x,y,z) \mid 1 < x < 3,\ y \leq 0,\ z \geq 0,\ \text{and } y+z > 0\},$$

where the two vertical half-lines from q_1 *and* q_2*, together with the enclosed face and the two lateral faces parallel to the* yz*-plane, would be erroneously excluded from the solution.* ∎

The example suggests that the generalization of Equation 10 fails when the convex polyhedron Q is not relatively open. Indeed, in the following we show that the natural generalization works when Q is relatively open, and then we provide a way to solve the problem for the general case.

For the first step we shall need a preliminary result stating that, whenever Q is contained in the topological closure of R, every point in the relative interior of Q can reach every point in $wedge_Q(R)$ while passing through R.

Lemma 9. *Let* Q *and* R *be two convex polyhedra such that* $Q \subseteq cl(R)$*. For all* $y \in wedge_Q(R)$ *and* $z \in rint(Q)$ *there exist* $d \in R \oplus \{-z\}$ *and* $\gamma \geq 0$ *such that* $y = z + \gamma\, d$*.*

Proof. By assumption, $y = x + \delta\, c$, for some $x \in Q$, $\delta \geq 0$ and $c \in R \oplus \{-x\}$. Let $c = r - x$, with $r \in R$. Since $x \in Q$ and $z \in rint(Q)$, there must be a point $x' \in Q$ along the line connecting x and z and such that z lies strictly between x and x'. By the convexity of R, the triangle with vertices x, x' and r is contained in $cl(R)$ (see Figure 14), and the relative interior of the triangle is contained in R. Hence, there is a point r' lying in the relative interior of the triangle and on the line segment connecting z and y. The direction d required by the thesis is then $r' - z$. ∎

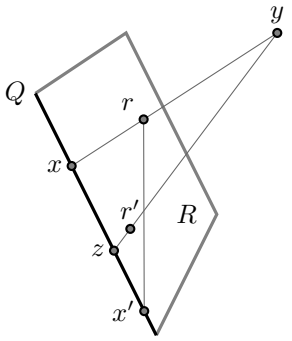


Figure 14: When $Q \subseteq cl(R)$, the points in $wedge_Q(R)$ can be reached from every point in $rint(Q)$ (see Lemma 9).

Thanks to the lemma above, we can prove that the generalization of Equation 10 is sound and complete for any relatively open convex polyhedron Q.

Lemma 10. *For all convex polyhedra* Q *and* R*, if* Q *is relatively open and* $Q \subseteq cl(R)$*, then it holds*

$$Cross(Q,R) = Q\swarrow_{>0} \cap\ mirror_Q(wedge_Q(R))\,. \tag{11}$$

Proof. [$\supseteq$] Let y be a point in r.h.s. of (11), we have that y can reach a point $z \in Q$ along a straight direction $c \in F$ in a positive amount of time $\delta_1 > 0$ (i.e., $z = y + \delta_1 c$). In addition, $y \in mirror_Q(wedge_Q(R))$.

Let y' be $mirror_Q(y)$, which belongs to $wedge_Q(R)$ (see Fig. 15). Assume, first, that y, z and y' identify a unique plane H (i.e., the three points are not collinear). Let x be the middle point of the segment with endpoints y and y'. Clearly, x belongs to the affine hull of Q as well as z, since $ahull(Q) \subseteq Q$. As a consequence, the line l connecting x and z is contained in $ahull(Q)$. We show that there is a neighborhood α of z that is contained both in l and in Q.

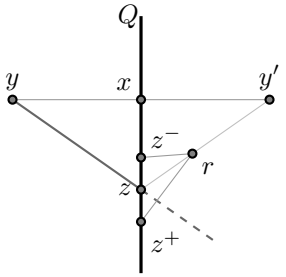


Figure 15: When a point y is in $Cross(Q,R)$, it can cross into R via Q following a single admissible direction (see Lemma 10).

Being $Q = rint(Q)$, there is a relative neighborhood of z entirely contained in Q. Hence, line l passing through z must intersect such a relative neighborhood, which, in turn, is the required neighborhood α of z. By Lemma 9, there exist $r \in R$ and $\delta \geq 0$ such that $y' = z + \delta(r - z)$. Now, let z^- and z^+ be two points, on either side of z along line l and contained in Q. The triangle with vertices z^-, r and z^+ is contained in $cl(R)$; the relative interior of the triangle is contained in R. The intersection of the relative interior of the triangle and the continuation of line yz contains a segment of positive length. Since all the points on that segment are contained in R and are reachable from y along a single direction $c \in F$, passing through $z \in Q$, we conclude that $y \in \mathit{Cross}(Q, R)$.

[$\subseteq$] Assume now $y \in \mathit{Cross}(Q, R)$. By definition, there exist $c \in F$, $0 < \delta_1 < \delta_2$, such that $z \triangleq y + \delta_1 c \in Q$ and $r \triangleq y + \delta_2 c \in R$. Let also $y' = \mathit{mirror}_Q(y)$ and x be the middle point of the segment with endpoints y and y'. Clearly, $y \in Q\swarrow_{>0}$. We need to prove that $y' \in \mathit{wedge}_Q(R)$. By a similar argument as in the previous case, there exist two points z^- and z^+, on either side of z along the line xz, that are contained in Q. The relative interior of the triangle with vertices z^-, r and z^+ is contained in R. The intersection of the above triangle and line zy' contains a segment of positive length (see again Fig. 15). Therefore, y' belongs to $\mathit{wedge}_Q(R)$ and, consequently, point y, the mirror of y' w.r.t. Q, belongs to $\mathit{Cross}(Q, R)$. ∎

It is now easy to compute $\mathit{Cross}(Q, R)$ a general convex polyhedron Q which is not relatively open. First, we notice that *Cross* distributes over unions in its first argument, i.e., $\mathit{Cross}(Q_1 \cup Q_2, R) = \mathit{Cross}(Q_1, R) \cup \mathit{Cross}(Q_2, R)$. Second, any convex polyhedron Q can be recursively partitioned into a finite set $\mathit{RelOpen}(Q)$ of relatively open convex polyhedra, as follows. Let $\mathit{RelOpen}(\emptyset) = \{\emptyset\}$, and

$$\mathit{RelOpen}(Q) = \{\mathit{rint}(Q)\} \cup \bigcup_{Q' \in [\![Q \setminus \mathit{rint}(Q)]\!]} \mathit{RelOpen}(Q').$$

The above recursion is well founded, since the affine dimension of every $Q' \in [\![Q \setminus \mathit{rint}(Q)]\!]$ is strictly lower than the affine dimension of Q, down to the case where Q' contains a single point, in which case Q' is relatively open and the recursion terminates. Hence, for all convex polyhedra Q and R, it holds

$$\mathit{Cross}(Q, R) = \bigcup_{Q' \in \mathit{RelOpen}(Q)} \mathit{Cross}(Q', R). \tag{12}$$

If, additionally, $Q \subseteq cl(R)$, by Lemma 10 we can compute each $\mathit{Cross}(Q', R)$ above, provided we can compute $\mathit{wedge}(\cdot)$, which is the topic of the following subsection.

5.2.5. Computing wedge

The following straightforward proposition states that if a convex polyhedron satisfies a linear inequality and its relative interior touches the corresponding hyperplane, then the whole polyhedron lies on the hyperplane.

Proposition 2. *For all convex polyhedra A and constraints $\alpha \triangleq a\,x \leq b$, if A is contained in the halfspace α and there exists $z \in \mathit{rint}(A)$ lying on the hyper plane $\beta \triangleq a\,x = b$ (i.e., $a\,z = b$), then A is contained in β.*

The following lemma shows that $\mathit{wedge}_A(B)$ is a convex polyhedron that can easily be computed from the representations of A and B. In particular, it is obtained by selecting a subset of the constraints defining B.

Lemma 11. *For all non-empty convex polyhedra A and B, such that $A \subseteq cl(B)$, the set $\mathit{wedge}_A(B)$ is the convex polyhedron defined as follows. Let $\{a_i x \sim_i b_i \mid i = 1, \ldots, k\}$ be a constraint system for B, where $\sim_i \in \{<, \leq\}$. Let $I \subseteq \{1, \ldots, k\}$ be the set of indices i such that all points in A satisfy $a_i x = b_i$. Then,*

$$\mathit{wedge}_A(B) = \bigcap_{i \in I} a_i x \sim_i b_i, \tag{13}$$

where the intersection over an empty I is taken to mean $\mathbb{R}^n$.

PROOF. *(⊆)* Let $y \in wedge_A(B)$, and let $u \in A$, $v \in B$, and $\delta \geq 0$ be such that $y = u + \delta(v - x)$. Let $i \in I$, we prove that $a_i y \sim_i b_i$. Our assumptions imply that $a_i u = b_i$ and $a_i v \sim_i b_i$. Then, $a_i y = a_i u + \delta(a_i v - a_i u) \sim_i b_i + \delta(b_i - b_i) = b_i$.

(⊇) Let y be a point belonging to the r.h.s. of (13). If $y \in B$, the thesis is immediate by setting $\delta = 1$ and $c = y - x$ as witness direction in the definition of *wedge*. Otherwise, let u be any point in the relative interior of A, and consider the line segment l connecting u and y. We prove that there exists a point $v \neq u$ on l that belongs to B. Assume the contrary. Then, each point on l, except possibly u, violates at least one of the constraints defining B. Hence, there is a constraint $a_i x \sim_i b_i$ among those that define B that is violated by points that are arbitrarily close to u. However, since $A \subseteq cl(B)$, $a_i u = b_i$. In words, the relative interior of A touches the hyperplane $a_i x = b_i$. Moreover, A is entirely contained in the halfspace $a_i x \leq b_i$. By Proposition 2, A is entirely contained in the hyperplane $a_i x = b_i$, and therefore $i \in I$. Summarizing, u lies on $a_i x = b_i$, and some points on l violate $a_i x \sim_i b_i$. As a consequence, y also violates $a_i x \sim_i b_i$, which is a contradiction because y belongs to the r.h.s. of (13). We conclude that there is a point $v \neq u$ on l belonging to B. Then, let $\delta = \frac{|y-u|}{|v-u|}$, we obtain $y = u + \delta v$ and the thesis. ∎

We can now provide the main result of the paper. Indeed, from Lemma 11, Lemma 10, and Equation 12, we obtain that the *Cross* operator is computable and, as a consequence of Equation 9, so is $CExt_3$. Theorems 5 and 6 provide the same result for the operators Ext_2 and Ext_3. Hence, by Lemma 6, we have that the extension of any convex type, and, therefore, of every canonical type, is computable. Finally, by Theorem 4 and by recalling that $RWA^{\mathcal{C}_\mathrm{d}}(G_1 \cup G_2, A)$ is equal to the union of $RWA^{\mathcal{C}_\mathrm{d}}(G_1, A)$ and $RWA^{\mathcal{C}_\mathrm{d}}(G_2, A)$, we obtain the desired result.

Theorem 7. *For all disjoint polyhedra G and A, $RWA^{\mathcal{C}_\mathrm{d}}(G, A)$ is computable.*

6. Computing RWA One Layer at a Time

In some sense, the procedure outlined at the end of Section 5 represents a depth-first computation of *RWA*. In this section we recall the alternative algorithm that we presented in the preliminary version of this work [8]. Such algorithm can be thought of as a breadth-first computation of *RWA*. Rather than computing the extension of a single type, the algorithm maintains a generally non-convex polyhedron W, containing all points that have already been proved members of *RWA*, regardless of their type, and it progressively enlarges W by invoking $Ext_2(P, P')$ and $Ext_3(P, \hat{P}, P')$, where P' is a convex polyhedra already in W, whereas P and $\hat{P}$ are patches of $\overline{A}$, i.e., sets of points that could belong to *RWA*. We conjecture that the breadth-first algorithm is generally more efficient than the depth-first one, since it involves fewer calls to the expensive Ext_3 operator. Indeed, our implementation, presented in Section 7, is based on this breadth-first version.

Analogously to the τ_ae operator, define the following:

$$\tau_\mathrm{d}(G, A, W) = G \cup \bigcup_{P \in [\![\overline{A}]\!]} Ext_3(P, G, \mathbb{R}^n) \cup \bigcup_{P, \hat{P} \in [\![\overline{A}]\!]} \bigcup_{P' \in [\![W]\!]} Ext_2(P, P') \cup Ext_3(P, \hat{P}, P').$$

A finite number of repeated applications of $\tau_\mathrm{d}(G, A, W)$, starting from $W = \emptyset$, captures exactly all points in $RWA^{\mathcal{C}_\mathrm{d}}(G, A)$. In particular, comparing the definition of τ_d with Equation 7, the first disjunct G corresponds to $Ext(G^0)$. The second disjunct collects the extensions of the types of the form $P^{>}G^0$, including the case $G^{>}G^0$, which is equivalent to type $G^{>}$. Finally, the last disjunct covers the two recursive cases of Equation 7, where P' ranges over the patches of already computed type extensions. Based on this correspondence, the following theorem can be proved.

Theorem 8 ([8]). *For all polyhedra A and convex polyhedra $G \in [\![\overline{A}]\!]$, we have*

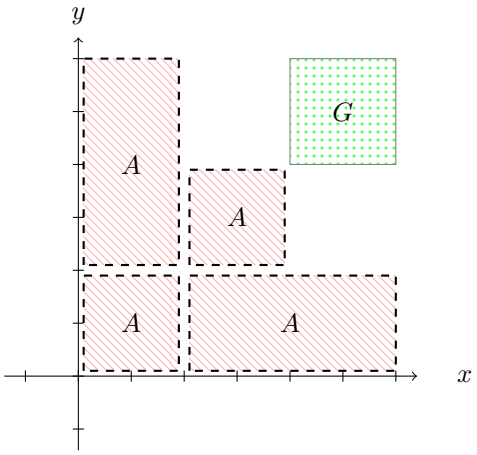
$$RWA^{\mathcal{C}_\mathrm{d}}(G, A) = \mu W \,.\, \tau_\mathrm{d}(G, A, W).$$

Moreover, the fixpoint is reached in a finite number of iterations.

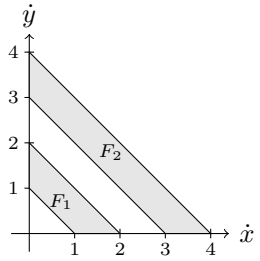
7. Experiments

In this section we report the results of experiments performed on a prototype implementation of the algorithm reported in Section 6, based on the Parma Polyhedra Library [15].

The input system is a variation of the 2-ship example presented in the Introduction: two ships must reach a target area while avoiding obstacles and keeping a minimum safety distance. Figure 16a shows the target area and the four obstacles. The latter are topologically open and weakly adjacent, so that only two 0-width channels are left between them. As advocated in the Introduction, the 0-width channels and the



(a) Target area G and avoidance region A.



(b) Range of velocities for the two ships.

Figure 16: The inputs to the 2-ship example.

differentiability constraint together provide an abstract way to model the fact that the ships cannot change direction at the intersection of the two channels (point $(2, 2)$). In practice, the ships may not be able to change direction due to a non-holonomic constraint (i.e., they may be going too fast to make the turn), or because their shapes and sizes prevent it (i.e., they are too large compared to the actual channel width).

Figure 16b shows the range of velocities (a.k.a. flow constraints) allowed for the two ships. The grayed area denoted by F_1 (resp., F_2) is the set of possible velocities for ship 1 (resp., 2). Both ships can move in all directions from 0° (i.e., straight along the x axis) to 90° (straight along the y axis), but they have a different range of speeds, with ship 1 being slower than ship 2. Neither of them can stop.

The resulting input system has 4 variables (x_1, y_1, x_2, y_2), representing the position of the two ships. The goal G consists in both ships being in the target area at the same time, as defined by $(x_1, y_1, x_2, y_2) \in [4, 6]^4$. The avoidance region A is the union of 9 convex polyhedra: 8 of them represent the collision of each ship with one of the 4 obstacles; the last polyhedron is the safety distance constraint, collecting all configurations in which the Chebyshev distance between the ships is at most 1:

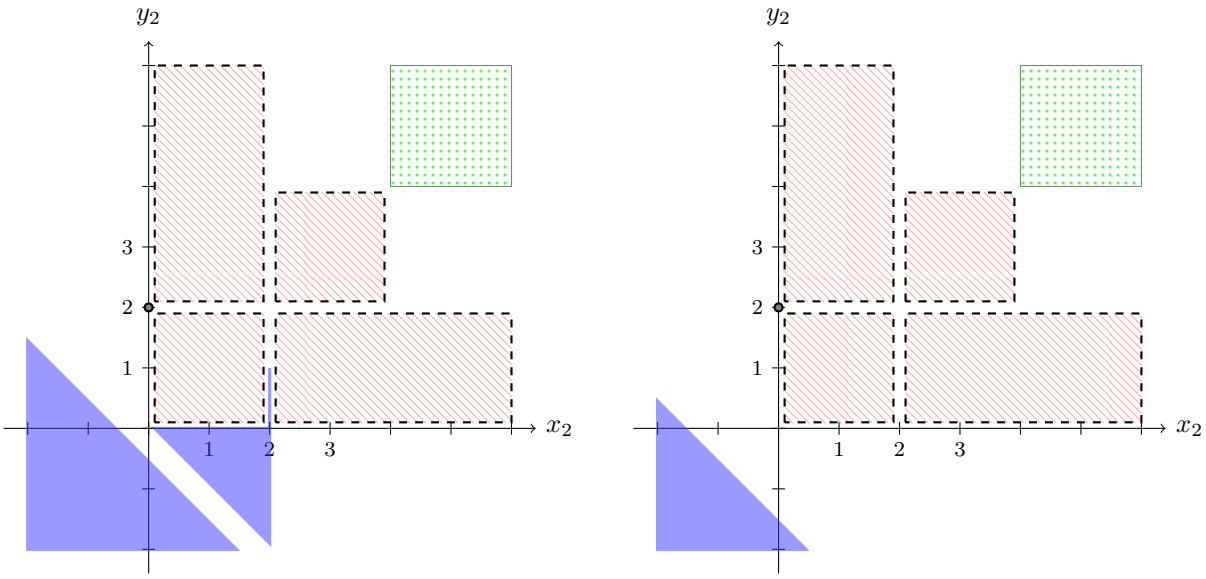
$$x_1 - x_2 \leq 1 \wedge x_2 - x_1 \leq 1 \wedge y_1 - y_2 \leq 1 \wedge y_2 - y_1 \leq 1.$$

Intuitively, this constraint sets an imaginary *safety barrier* around each ship, in the shape of an axis-aligned square of size 2, centered on each ship, which the other ship cannot enter.

We used our tool to compute the sets $RWA^{\mathcal{C}}(G, A)$, for the two classes of trajectories $\mathcal{C} \in \{\mathcal{C}_{\text{ae}}, \mathcal{C}_{\text{s}}\}$[3]. It is worth stressing that these sets represent the exact solution to the controllability problem, and cover all possible initial positions of the two ships.

Figure 17 shows in blue the slice of the solutions when ship 1 is located at $(0, 2)$. The solution for smooth trajectories (Fig. 17b) is significantly smaller than the one for a.e. differentiable trajectories (Fig. 17a). To

[3]The results were obtained in ∼12 seconds for $\mathcal{C}_{\text{ae}}$ and ∼320 seconds for $\mathcal{C}_{\text{s}}$ on a 1.7Ghz Intel Core i7.



(a) Almost everywhere differentiable trajectories.

(b) Smooth trajectories.

Figure 17: Solution slice with $x_1 = 0, y_1 = 2$ in the two semantics.

investigate this difference, consider starting point $(2, -1)$ for ship 2. This point belongs to the solution for $\mathcal{C}_{\text{ae}}$ but not for $\mathcal{C}_{\text{s}}$.

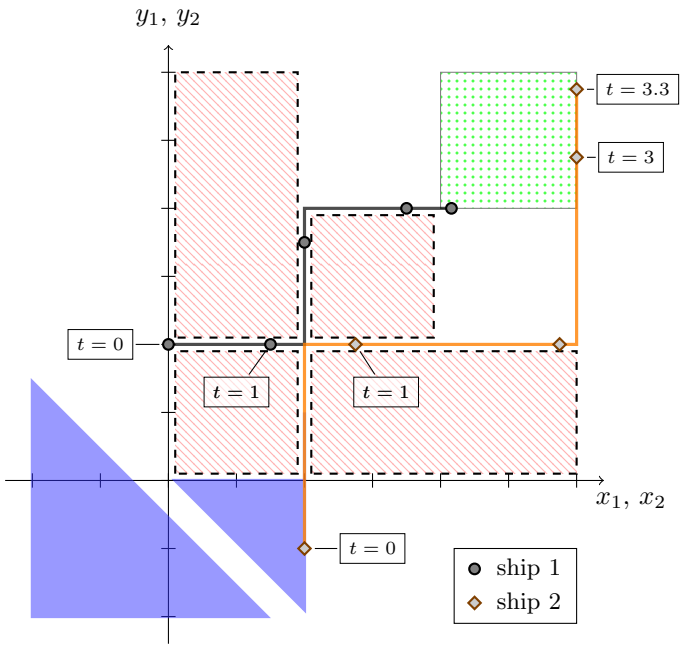
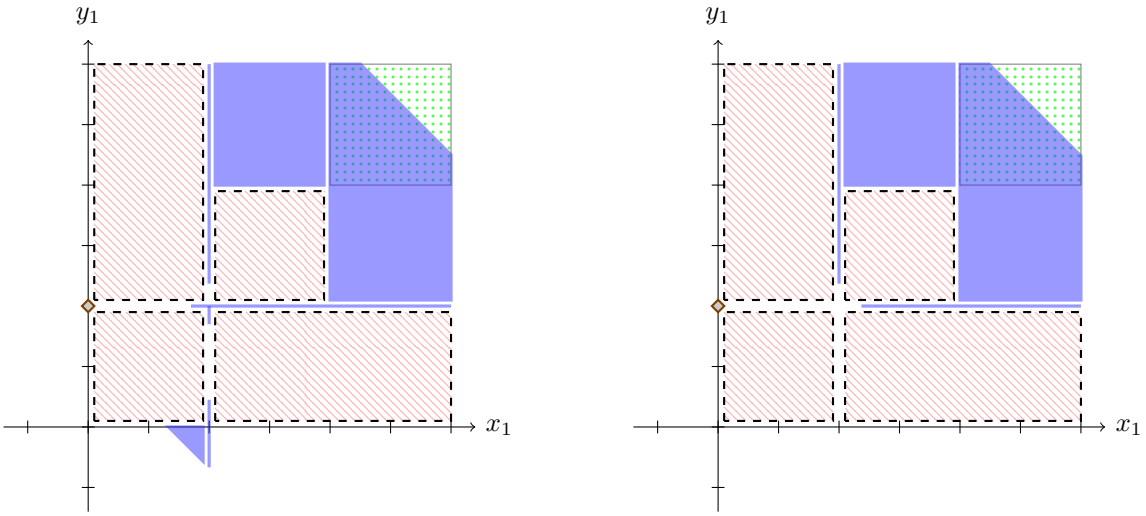


Figure 18: Witness trajectories with the a.e. differentiable semantics. Ship positions are marked at times $t = 0, 1, 2, 3, 3.3$ (best seen in color).

To see why that configuration is only part of the solution if the ships are allowed to change direction at the intersection, consider the witness in Figure 18. The two thick lines represent the trajectories for the two ships. Moreover, the position of the vessels is marked for times $t \in \{0, 1, 2, 3, 3.3\}$, with 3.3 being a time in which the ships are simultaneously in the target area. In the first time unit, ship 2 quickly reaches the intersection point $(2, 2)$ and then changes direction eastwards. In the meanwhile, ship 1 slowly reaches the intersection and then turns northwards. In this way, they manage to go though the intersection while keeping the minimum required distance. On the other hand, consider the smooth semantics, in which the ships cannot change direction instantaneously. In that case, once ship 2 reaches the intersection, it is forced to continue northwards. Then, even at its maximum speed, it cannot escape fast enough from the safety barrier of the approaching ship 1.

Next, consider the slice of the solutions when we fix the position of ship 2 to the point $(0, 2)$ (see Figure 19). In both semantics, if the slower ship 1 has already cleared the intersection, the faster ship 2 can catch up and reach the target area in time. On the other hand, when ship 1 starts behind the intersection (say, at point

$(2, 0)$), the two semantics exhibit a significant difference. Broadly speaking, in the smooth semantics ship 2 must initially move at its fastest speed for a longer period of time, in order to avoid a collision with ship 1. Afterwards, even if ship 2 slows down and ship 1 speeds up, they will not be able to be in the target area at the same time. This explains why points such as $(2, 0)$ belong to the solution for the a.e. differentiable semantics (Fig. 19a) but not in the smooth semantics (Fig. 19b).



(a) Almost everywhere differentiable trajectories.

(b) Smooth trajectories.

Figure 19: Solution slice with $x_2 = 0, y_2 = 2$ in the two semantics.

8. Conclusions

In this paper we considered the problem of computing the set of points that can reach a given polyhedron while avoiding another one via a differentiable or smooth trajectory that is subject to a polyhedral differential inclusion. This problem is relevant for the symbolic analysis of continuous time dynamic systems such as Linear Hybrid Automata.

We have shown that previous solutions do not guarantee differentiability of the trajectories, particularly because it is not sufficient to consider pairs of adjacent polyhedra when extending the set of possible trajectories. Rather, triples of *weakly adjacent* polyhedra come into play.

We provided a precise formulation of the problem, allowing us to prove that the distinction between differentiable and smooth trajectories is immaterial to it, as both constraints lead to the same result. We then presented an exact symbolic algorithm for the problem, which works in the original state space (i.e., it does not add extra dimensions) and is based on reflections of polyhedra. Our algorithm uses only standard operations on polyhedra that are offered by state-of-the-art libraries such as Parma Polyhedra Library [15].

The implementation used to carry out the experiments in Section 7 is a preliminary proof-of-concept. As future work, we plan to explore performance improvements, based, for instance, on polyhedra adjacency, in the same vein as we have done in previous work [9]. This could, in principle, mitigate the performance gap with respect to the a.e. differentiable semantics. We also plan to integrate the algorithm into existing tools for verification of hybrid systems, such as SpaceEx [16].

References

[1] R. Alur, T. Henzinger, P.-H. Ho, Automatic symbolic verification of embedded systems, IEEE Trans. Softw. Eng. 22 (1996) 181–201. doi:10.1109/32.489079.

[2] T. Henzinger, P.-H. Ho, H. Wong-toi, Algorithmic analysis of nonlinear hybrid systems, IEEE Transactions on Automatic Control 43 (1998) 540–554.

[3] C. Tomlin, J. Lygeros, S. Shankar Sastry, A game theoretic approach to controller design for hybrid systems, Proc. of the IEEE 88 (7) (2000) 949–970.

[4] A. Balluchi, L. Benvenuti, T. Villa, H. Wong-Toi, A. Sangiovanni-Vincentelli, Controller synthesis for hybrid systems with a lower bound on event separation, Int. J. of Control 76 (12) (2003) 1171–1200. doi:10.1080/0020717031000123616.

[5] M. Benerecetti, M. Faella, S. Minopoli, Reachability games for linear hybrid systems, in: HSCC 12: Hybrid Systems Computation and Control. 15th Int. Conf., ACM, 2012, pp. 65–74.

[6] M. Benerecetti, M. Faella, Automatic synthesis of switching controllers for linear hybrid systems: Reachability control, ACM Trans. Embed. Comput. Syst. 16 (4) (2017) 104:1–104:27. doi:10.1145/3047500.

[7] T. Henzinger, B. Horowitz, R. Majumdar, Rectangular hybrid games, in: CONCUR 99: Concurrency Theory. 10th Int. Conf., Vol. 1664 of Lect. Notes in Comp. Sci., Springer, 1999, pp. 320–335. doi:10.1007/3-540-48320-9_23.

[8] M. Benerecetti, M. Faella, Tracking differentiable trajectories across polyhedra boundaries, in: HSCC 13: Hybrid Systems Computation and Control. 16th Int. Conf., ACM, 2013, pp. 193–202. doi:10.1145/2461328.2461360.

[9] M. Benerecetti, M. Faella, S. Minopoli, Automatic synthesis of switching controllers for linear hybrid systems: Safety control, Theoretical Computer Science 493 (2013) 116–138.

[10] M. Benerecetti, M. Faella, S. Minopoli, Towards efficient exact synthesis for linear hybrid systems, in: GandALF 11: 2nd Int. Symp. on Games, Automata, Logics and Formal Verification, Vol. 54 of Electronic Proceedings in Theoretical Computer Science, 2011. doi:10.4204/EPTCS.54.19.

[11] N. Halbwachs, Y.-E. Proy, P. Roumanoff, Verification of real-time systems using linear relation analysis, Formal Methods in System Design 11 (1997) 157–185. doi:10.1023/A:1008678014487.

[12] M. Benerecetti, M. Faella, S. Minopoli, Revisiting synthesis of switching controllers for linear hybrid systems, in: Proc. of the 50th IEEE Conf. on Decision and Control, IEEE, 2011.

[13] R. Alur, T. Henzinger, P.-H. Ho, Automatic symbolic verification of embedded systems, in: RTSS 93: Real-Time Systems Symposium, 1993, pp. 2–11.

[14] R. Bagnara, P. M. Hill, E. Zaffanella, Not necessarily closed convex polyhedra and the double description method, Formal Aspects of Computing 17 (2005) 222–257. doi:10.1007/s00165-005-0061-1.

[15] R. Bagnara, P. M. Hill, E. Zaffanella, The Parma Polyhedra Library: Toward a complete set of numerical abstractions for the analysis and verification of hardware and software systems, Science of Computer Programming 72 (1–2) (2008) 3–21. doi:10.1016/j.scico.2007.08.001.

[16] G. Frehse, C. L. Guernic, A. Donzé, S. Cotton, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, O. Maler, Spaceex: Scalable verification of hybrid systems, in: CAV 11: Proc. of 23rd Conf. on Computer Aided Verification, 2011, pp. 379–395.

Appendix A. Additional Proofs

Lemma 1 (Tangent). *Let P be a convex polyhedron, x a point, $f \in Adm^{\mathcal{C}_d}(x)$ a trajectory, and $\hat{\delta} > 0$ a delay such that in all non-empty intervals $(\delta, \hat{\delta})$ there is a time γ such that $f(\gamma) \in P$. Then, there exists $\delta^* > 0$ such that $f(\hat{\delta}) - \delta^* \dot{f}(\hat{\delta}) \in cl(P) \cap \{x\} \nearrow_{>0}$.*

PROOF. Let $y = f(\hat{\delta})$ and $c = \dot{f}(\hat{\delta})$. Notice that $y \in cl(P)$ because in all neighborhoods of y there is a point in P. Moreover, by Proposition 1 all points of the trajectory belong to $\{x\} \nearrow_{>0}$.

Assume by contradiction that for all $\delta > 0$ it holds $y - \delta c \notin cl(P) \cap \{x\} \nearrow_{>0}$. In this case, the half-line $\{y - \delta c \mid \delta \geq 0\}$ intersects $Q \triangleq cl(P) \cap \{x\} \nearrow_{>0}$ only in point y. Since Q is a convex polyhedron, there is a linear constraint which is satisfied by y and violated by all other points of the half-line. Hence, the distance between Q and a generic point $y - \delta c$ of the half-line grows at least linearly with δ. Formally, there exists $a > 0$ such that $dist(y - \delta c, Q) \geq \delta a$ for all $\delta \geq 0$, where $dist$ denotes the Euclidean distance.

On the other hand, by definition $\lim_{\delta \to 0} \frac{y - f(\hat{\delta} - \delta)}{\delta} = c$. Therefore, there exists $\delta^* > 0$ such that for all $0 < \delta < \delta^*$ it holds

$$\left\| \frac{y - f(\hat{\delta} - \delta)}{\delta} - c \right\| < a, \text{i.e.}$$
$$\|(y - \delta c) - f(\hat{\delta} - \delta)\| < \delta a$$
$$dist(y - \delta c, f(\hat{\delta} - \delta)) < \delta a.$$

As a consequence, we have that $f(\hat{\delta} - \delta) \notin Q$, contradicting the assumption that in all left-neighborhoods of $\hat{\delta}$ there is a time when f lies in P. ∎

Claim 1. *Given the interval $[0, \bar{\delta}]$ and $\delta_1 \geq 0$, there exists an $\alpha > 0$ such that the function g, defined by equation (6), with parameter α satisfies all the conditions (1)–(5).*

PROOF. It is easy to verify that $g(\delta)$ is continuous in the interval $[0, \bar{\delta}]$ and satisfies conditions (1) and (2) by construction. Moreover, since $LF(x) \in (0, 1)$, for all $x \in \mathbb{R}$, it immediately follows that $g(\delta) \in (0, 1)$, for all $\delta \in [0, \bar{\delta}]$, thus satisfying condition (3). When we want to emphasize the dependency of h and g on the parameter α, we will write h_α and g_α. Figure A.20 shows three instances of g_α, for different values of α. Regarding property (4), the first derivative of $g(\delta)$ is:

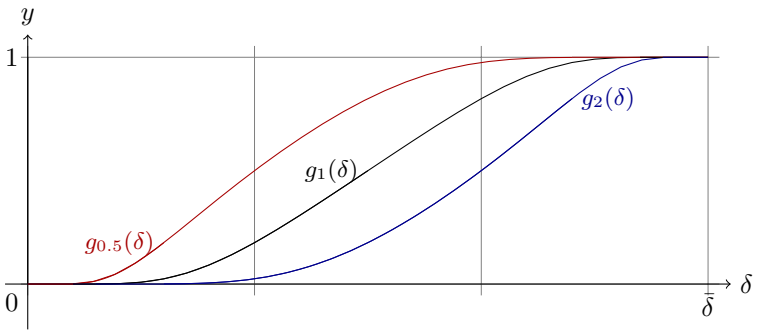


Figure A.20: Three instances of $g_\alpha(\delta)$, for $\alpha = 0.5$, $\alpha = 1$ and $\alpha = 2$.

$$\frac{d}{d\,\delta} g(\delta) = \frac{1}{1 + e^{-h(\delta)}} \frac{e^{-h(\delta)}}{1 + e^{-h(\delta)}} \dot{h}(\delta) = g(\delta)\,(1 - g(\delta))\ \dot{h}(\delta). \tag{A.1}$$

In addition, one can easily prove by induction on $k \geq 1$ that

$$\frac{d^k}{d\delta^k} h(\delta) = \bar{\delta}\, k! \left[\frac{1}{\alpha(\bar{\delta} - \delta)^{k+1}} - \frac{(-1)^k \alpha}{\delta^{k+1}} \right]. \tag{A.2}$$

Hence, it is immediate to see that the k-th derivative of $h(\delta)$ w.r.t. δ is the ratio between two polynomials in δ, with the degree of the denominator greater than that of the numerator. Moreover, the limits at 0 and $\bar{\delta}$ of the ratio is 0.

Proposition 3. *The k-th derivative of $g(\delta)$ w.r.t. δ is of the form* [4]

$$\sum_i a_i \cdot g^{n_i}(\delta) \cdot (1 - g(\delta))^{m_i} \cdot \frac{p_i(\delta)}{q_i(\delta)}, \tag{A.3}$$

for suitable integer constants a_i, $n_i \geq 1$, $m_i \geq 1$, and for $p_i(\delta)$ and $q_i(\delta)$ polynomials in δ with $deg(p_i(\delta)) < deg(q_i(\delta))$.

PROOF. The base case for $k = 1$ is given by equation (A.1). For the inductive case, it suffices to observe that the derivative of each term in the summation above reduces to the summation of three additional terms, each of the form $a \cdot g^n(\delta) \cdot (1 - g(\delta))^m \cdot \frac{p_j(\delta)}{q_j(\delta)}$, for suitable a, n, and m. Indeed,

$$\begin{aligned}\frac{d}{d\delta}\left(a_i \cdot g^{n_i}(\delta) \cdot (1 - g(\delta))^{m_i} \cdot \frac{p_i(\delta)}{q_i(\delta)}\right) = &- a_i \cdot n_i \cdot g^{n_i}(\delta) \cdot (1 - g(\delta))^{m_i+1} \cdot \dot{h}(\delta) \cdot \frac{p_i(\delta)}{q_i(\delta)} + \\ &+ a_i \cdot m_i \cdot g^{n_i+1}(\delta) \cdot (1 - g(\delta))^{m_i} \cdot \dot{h}(\delta) \cdot \frac{p_i(\delta)}{q_i(\delta)} + \\ &+ a_i \cdot g^{n_i}(\delta) \cdot (1 - g(\delta))^{m_i} \cdot \frac{p_i(\delta)\dot{q}_i(\delta) + \dot{p}_i(\delta)q_i(\delta)}{q_i^2(\delta)}.\end{aligned}$$

It is immediate to see that each quotient of polynomials in all the terms still satisfies the required condition that the degree of the numerator is smaller than the one of the denumerator *(end of Proposition proof).*

Due to Proposition 3, in order to prove that $g^{(k)}(0) = 0$ it is sufficient to show that for all polynomials $p(\delta)$ and $q(\delta)$, and all integers $a, n \geq 1, m \geq 1$, it holds:

$$\lim_{\delta \to 0^+} a \cdot g^n(\delta) \cdot (1 - g(\delta))^m \cdot \frac{p(\delta)}{q(\delta)} = 0.$$

Notice that in any bounded neighborhood of 0 the polynomial p is bounded and so is its limit for $\delta \to 0^+$. Moreover, the limit of the term $(1 - g(\delta))^m$ for $\delta \to 0^+$ is 1, due to condition (1). We are left to study the limit of the quotient $\frac{g^n(\delta)}{q(\delta)}$, which is non-trivial if $q(0) = 0$.

We distinguish three cases, based on the position of α w.r.t. 1. If $\alpha > 1$ then we prove that when δ is sufficiently close to 0, it holds that

$$g(\delta) = \frac{1}{1 + e^{-h(\delta)}} < \frac{1}{1 + e^{\frac{1}{\delta}}}. \tag{A.4}$$

In other words, $-h(\delta) > \frac{1}{\delta}$. Assume for simplicity that $\bar{\delta} = 1$, we have the following:

$$-h(\delta) = \frac{\alpha(1 - \delta)}{\delta} - \frac{\delta}{\alpha(1 - \delta)}.$$

Hence, we should verify

$$\begin{aligned}\alpha^2(1 - \delta)^2 - \delta^2 &> \alpha(1 - \delta) \\ \alpha^2(1 - \delta)^2 - \alpha(1 - \delta) - \delta^2 &> 0.\end{aligned}$$

Since

$$\lim_{\delta \to 0^+} \alpha^2(1 - \delta)^2 - \alpha(1 - \delta) - \delta^2 = \alpha^2 - \alpha > 0,$$

[4] Note that $1 \leq i \leq 3^{k-1}$.

we obtain the thesis (A.4). We then go back to studying the ratio between $g(\delta)^n$ and $q(\delta)$ when δ approaches 0:

$$\begin{aligned}
\lim_{\delta\to 0^+} \frac{g(\delta)^n}{q(\delta)} &\leq \lim_{\delta\to 0^+} \frac{g(\delta)}{q(\delta)} \\
&\leq \lim_{\delta\to 0^+} \frac{1}{1+e^{\frac{1}{\delta}}} \cdot \frac{1}{q(\delta)} && \text{by (A.4)} \\
&\leq \lim_{\delta\to 0^+} \frac{1}{e^{\frac{1}{\delta}}} \cdot \frac{1}{q(\delta)} \\
&= \lim_{\delta\to 0^+} \frac{e^{-\frac{1}{\delta}}}{q(\delta)} \\
&= \lim_{y\to -\infty} \frac{e^y}{q(-\frac{1}{y})} \\
&= \lim_{y\to -\infty} \frac{y^l \cdot e^y}{q'(y)} && \text{where } l = \deg(q) \text{ and } q' \text{ is a polynomial} \\
&= 0.
\end{aligned}$$

Next, we consider the case $\alpha < 1$. Similarly to the previous case, we observe that when δ is sufficiently close to zero, it holds that

$$g(\delta) = \frac{1}{1+e^{-h(\delta)}} < \frac{1}{1+e^{\frac{\alpha^2}{\delta}}}.$$

In other words, $-h(\delta) > \frac{\alpha^2}{\delta}$. Hence,

$$\begin{aligned}
\lim_{\delta\to 0^+} \frac{g(\delta)^n}{q(\delta)} &= \lim_{\delta\to 0^+} \frac{e^{-\frac{\alpha^2}{\delta}}}{q(\delta)} \\
&= \lim_{y\to -\infty} \frac{e^y}{q(-\frac{\alpha^2}{y})} \\
&= \lim_{y\to -\infty} \frac{y^l \cdot e^y}{q'(y)} && \text{where } l = \deg(q) \text{ and } q' \text{ is a polynomial} \\
&= 0.
\end{aligned}$$

Finally, the case $\alpha = 1$ can be treated along similar lines, by observing that in a neighborhood of 0 it holds that

$$g(\delta) < \frac{1}{1+e^{\frac{1}{2\delta}}}.$$

This proves that $g^{(k)}(0) = 0$. The other half of property (4) (i.e., $g^{(k)}(\bar{\delta}) = 0$) can be proved in a symmetrical fashion.

We are left with condition (5), i.e., for each choice of δ_1, δ_2 there exists α such that the integral of g_α between 0 and $\bar{\delta}$ is δ_1. We show this result indirectly: we prove that by an appropriate choice of α, the integral can be made arbitrarily close to 0 or arbitrarily close to $\bar{\delta}$. The desired property then follows by continuity.

The following proposition allows us to prove that the integral can be made arbitrarily close to zero. As illustrated by Figure A.21, if an appropriate choice of α makes g_α pass *below* the point of coordinates $(\bar{\delta}-a, a)$, considering that g_α is monotonically increasing, this implies that the integral of g_α between 0 and $\bar{\delta}$ is at most equal to the area of the two rectangles of corners $(0,0)$-$(\bar{\delta}-a, a)$ and $(\bar{\delta}-a, 0)$-$(\bar{\delta}, 1)$, respectively (i.e., the shaded region in Figure A.21). If this property holds for an arbitrary $a > 0$, said integral can be made arbitrarily small.

Proposition 4. *For all $a > 0$ there exists $\alpha > 0$ such that $g_\alpha^{-1}(a) > \bar{\delta} - a$.*

Notice that it holds $g_\alpha^{-1}(y) = h_\alpha^{-1}(LF^{-1}(y))$. The inverse functions LF^{-1} and h_α^{-1} can easily be computed and are expressed as follows:

$$LF^{-1}(y) = -\log\left(\frac{1}{y} - 1\right)$$

and

$$h_\alpha^{-1}(y) = \alpha\bar{\delta} \cdot \frac{y - 2\alpha + \sqrt{y^2 + 4}}{2(1 + \alpha y - \alpha^2)}. \tag{A.5}$$

Now, $LF^{-1}(y)$ maps the interval $(0, 1)$ onto $\mathbb{R}$ and $h^{-1}(y)$ maps $\mathbb{R}$ onto the interval $(0, \bar{\delta})$. In order to prove Proposition 4 it is sufficient to prove that for all $y \in \mathbb{R}$ it holds $\lim_{\alpha\to\infty} h_\alpha^{-1}(y) = \bar{\delta}$. The latter property can be verified by inspecting (A.5).

Suppose we want to prove that the integral can be made smaller than $\epsilon > 0$. Then, we apply Proposition 4 to $a = \dfrac{\epsilon}{\bar{\delta} + 1}$ and we obtain as a consequence that

$$\int_0^{\bar{\delta}} g(\delta)d\delta < \bar{\delta}a + a = \epsilon.$$

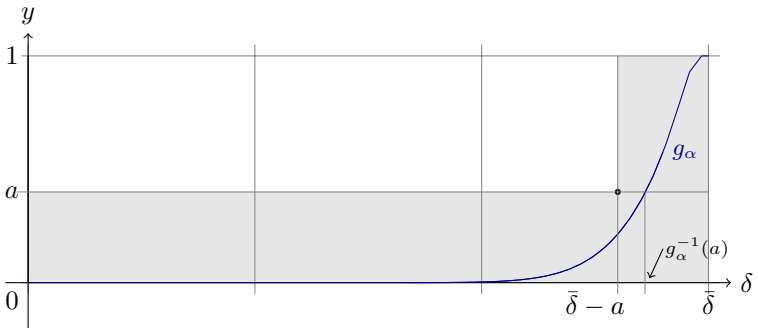


Figure A.21: Illustrating Proposition 4. The curve g_α can be made to pass below any point of the type $(\bar{\delta} - a, a)$, for all $a > 0$. Hence, its integral can be made arbitrarily small.

The argument for the fact that the integral can be made arbitrarily close to $\bar{\delta}$ is symmetrical. ∎

Theorem 6. *For all convex polyhedra P, $\hat{P}$ and P' the following holds:*

$$Ext_3(P, \hat{P}, P') = P \cap CExt_3(P, \hat{P}, P')_{\swarrow} .$$

Proof. *($\supseteq$)* This part is illustrated in Figure A.22. Let $x \in P \cap CExt_3(P, \hat{P}, P')_{\swarrow}$ and let $y \in CExt_3(P, \hat{P}, P')$ be such that $x \in \{y\}_{\swarrow}$. There exist $c \in F$ and $0 < \delta_1 < \delta_2$ satisfying the definition of $CExt_3(P, \hat{P}, P')$ for y.

Let $z = y + \delta_1 c$; by definition it holds $z \in \hat{P}$. Moreover, since $y + \delta c \in cl(P)$ for all $\delta \in (0, \delta_1)$, also $z \in cl(P)$ holds. By applying Lemma 2 with $x_0 = x$, $x_1 = y$ and $x_2 = z$, we get a differentiable trajectory f in $Adm^{\mathcal{C}_d}(x)$ from x to z whose derivative in z is c. In addition, f lies within P in all the points from x to z, except, possibly, z itself.

Let now $v = y + \delta_2 c$. By the definition of $CExt_3(P, \hat{P}, P')$, we have that $v \in cl(P') \cap P'_{\swarrow}$, therefore there exists a point $t \in P'$ reachable from v following a straight directon in F. We can then apply once again Lemma 2 with $x_0 = z$, $x_1 = v$ and $x_2 = t$, obtaining an admissible differentiable trajectory g whose derivative in z is c. Similarly to f, g lies within P' in all the points from z to t, except, possibly, z itself.

Therefore, the concatenation of f and g in z is differentiable everywhere, leads from x to a point in P', and satisfies all the requirements of $Ext_3(P, \hat{P}, P')$.

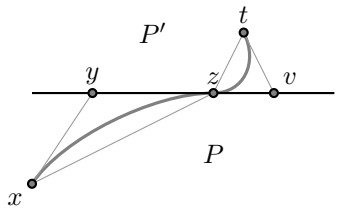


Figure A.22: When a point $x \in P$ can reach another point y which is in $CExt_{\mathcal{J}}(P, \hat{P}, P')$, x can differentiably pass into P' (see Theorem 6). Here, $\hat{P} = \{z\}$.

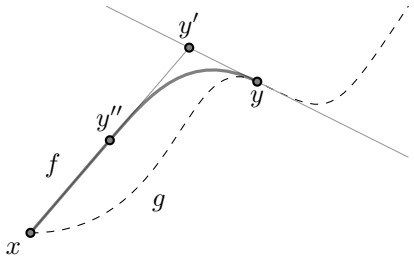


Figure A.23: Converting an arbitrary trajectory (dashed line) into one that starts with a straight segment (see Lemma 5).

($\subseteq$) Let $x \in Ext_{\mathcal{J}}(P, \hat{P}, P')$ and let $f \in Adm(x)$ and δ_1, δ_2 be the trajectory and the delays whose existence is postulated by the definition of $Ext_{\mathcal{J}}$. By definition, $x \in P$. Let $y = f(\delta_1)$, recall that $y \in \hat{P}$. Moreover, since $f(\delta) \in P$ for all $\delta \in [0, \delta_1)$, we also have that $y \in cl(P)$.

Let $c = \dot{f}(\delta_1) \in F$, by Lemma 1, there exists $\gamma_1 > 0$ such that $u_1 \triangleq y - \gamma_1 c \in cl(P) \cap \{x\}\nearrow$. Since $u_1 \in cl(P)$ and $y \in cl(P)$, then for all $\delta \in (0, \gamma_1)$ we have $u_1 + \delta c \in cl(P)$.

Similarly, by applying Lemma 1 backwards from a point $f(\delta) \in P'$ (it is sufficient to consider any $\delta \in (\delta_1, \delta_2)$), we obtain another value $\gamma_2 > 0$, such that $u_2 \triangleq y + \gamma_2 c \in cl(P') \cap P'\swarrow$. Since $y \in cl(P')$ and $u_2 \in cl(P')$, then for all $\delta \in (\gamma_1, \gamma_1 + \gamma_2)$ we have $y + \delta c \in cl(P')$. As a consequence, we obtain that $x \in P \cap CExt_{\mathcal{J}}(P, \hat{P}, P')_{\swarrow}$. ∎

Lemma 5. *For all types $\mathbf{T}$ and $x \in Ext(\mathbf{T})$ there exist a trajectory $f \in Adm(x)$ of type $\mathbf{T}$, a positive delay δ^*, and a slope $c \in F$ such that $\dot{f}(\delta) = c$ for all $\delta \in [0, \delta^*)$.*

Proof. If $\mathbf{T} = \varepsilon$ the thesis is trivial, because all trajectories have type $\mathbf{T}$ in the interval $[0, 0]$, including those trajectories that start with a straight segment. Otherwise, let $x \in Ext(\mathbf{T})$ and $g \in Adm^{\mathcal{C}^1}(x)$ be a trajectory of type $\mathbf{T}$. Let P be the first polyhedron occurring in $\mathbf{T}$, we distinguish two cases: (i) $\mathbf{T} = P^{>}\mathbf{T}'$, or (ii) $\mathbf{T} = P^{0}\mathbf{T}'$. The essential features of both cases are shown in Figure A.23.

(i) Since $\mathbf{T} = P^{>}\mathbf{T}'$, the trajectory g lingers in P for a positive amount of time. Let $\bar{\delta} > 0$ be a delay such that g lies in P at all times between 0 and $\bar{\delta}$.

Let $y = g(\bar{\delta})$ and $c = \dot{g}(\bar{\delta})$; by Lemma 1, there exists $\delta' > 0$ such that $y' \triangleq y - \delta' c \in cl(P) \cap \{x\}\nearrow_{>0}$. We build a new trajectory $f \in Adm^{\mathcal{C}^1}(x)$ as follows (see Figure A.23): starting from x, the trajectory f follows the straight line from x to y' up to an arbitrary intermediate point $y'' = f(\delta^*)$; then, we apply Lemma 2 to points $x_0 = y''$, $x_1 = y'$, $x_2 = y$, thus obtaining a curve that reaches y with final slope c. After reaching point y, the new trajectory f proceeds as the old one g. It is easy to verify that f is differentiable (in particular, in y'' and y). Moreover, f lies in P from time 0 to the time required to reach y, because it is contained in the triangle with vertices x, y', and y, of which x and y belong to P, and y' to $cl(P)$. After reaching y, f coincides with g, hence f also has type $\mathbf{T}$.

(ii) The trajectory g immediately exits from P. If $\mathbf{T}' = \varepsilon$, then the straight-line trajectory f with slope $\dot{g}(x)$ is the desired trajectory. Otherwise $\mathbf{T}' = Q^{>}\mathbf{T}''$ and g lingers in Q, i.e., there exists $\bar{\delta} > 0$ such that $g(\delta) \in Q$ for all $\delta \in (0, \bar{\delta})$. Notice that $x \in cl(Q)$. We build a new trajectory f in the same way as in the previous case. However, now we have $x \in cl(Q)$, and $y, y', y'' \in Q$. It follows that f lies in Q for all positive time delays up to the time required to reach y. After that time, f coincides with g. Hence, f has type $\mathbf{T}$. ∎