Approximating Viability Kernels of Non-linear Systems

André M. Carvalho and João S. Sequeira
Instituto Superior Técnico / Institute for Systems and Robotics, Lisbon, Portugal

Keywords: Viability Kernel, Reachability, Non-linear Systems.

Abstract: A typical concern in Robotics is to assess if it is possible to keep a robot inside a set of safe states, e.g., an autonomous car that must stay on the road. That is closely tied with the problem of computing the viability kernel of the system, i.e., the largest set of initial states for which it is guaranteed that the system has controls that keep maintain the trajectories inside the constraint set. The approach in this paper builds on previous work, on linear sampled-data systems. It is based on sampling the boundary of the constraint set, finding the states inside the viability kernel using finite-horizon forward simulation. Our adaptation extends the original algorithm, approximating the viability kernel for some non-linear systems through linearization methods. The non-linear systems here approached are the ones described by first order differential equations with continuous derivatives and convex with respect to the inputs. Existence and uniqueness conditions are also established to ensure adequate results for the whole algorithm. A practical example, with a simple non-linear system, to illustrate the proposed algorithm is also presented.

1 INTRODUCTION

This work is motivated by control problems in the field of autonomous vehicles, namely what is the admissible set of controls that keeps a vehicle on the road. The differential inclusions provide a generalization of the differential equations, commonly used to model dynamics of vehicles that (i) has similarities to human-like strategies to model navigation problems and (ii) embeds uncertainties in a very intuitive way. Viability theory(Aubin et al., 2011), models the navigation of autonomous systems very similar to the way a human driver processes information and acts. There exists a set of trajectories that are viable - keep the car on track, safely - instead of the usual tracking of a specific parametrized curve in the state space. The viability kernel is a central piece in this theory: it is the largest set of initial states, that starting in it, there exists at least one trajectory that is kept inside the constraint set.

The classical numerical method to compute the viability kernel for any type of system is based on griding the state space, in (Saint-Pierre, 1994). However, griding methods are known to have exponential complexity in the number of dimensions of the system. This method was applied in the context of autonomous driving in (Liniger and Lygeros, 2017).

Our work main procedure is based in the random supporting directions used in (Gillula et al., 2014). The most common representation for convex sets are polytopes and zonotopes (Girard et al., 2006). Both objects are closed under the operations used in the computation of reachable sets of linear systems, such as Minkowski sums and affine-maps. Polytopes have a major disadvantage when used in high dimensional systems(Tiwary, 2008). Zonotopes are more appealing than polytopes to compute reachable sets in these cases (Girard, 2005).

The original forward simulation is modified into a reachable set computation followed by testing if the reachable and the constraint set intersect each other. We prove that this modification yields the same, correct, result. Unlike the original work by (Gillula et al., 2014), our adaptation can be used in non-linear systems as long as the reachable set for that system can be computed and whether the overlapping test between two sets can be determined. We use a method to compute reachable sets of non-linear systems based on (Althoff, 2010). The overlapping test is here implemented as a simple Linear Programming optimization problem, without computing explicitly the intersection of both sets.

The paper is organized as follows. Section 2 presents background results. Section 3 describes the original algorithm and our adaptation to find viability kernels of non-linear systems. Section 4 presents re-

Carvalho, A. and Sequeira, J.
Approximating Viability Kernels of Non-linear Systems.
DOI: 10.5220/0006860903330340
In *Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2018) - Volume 2*, pages 333-340
ISBN: 978-989-758-321-6

sults on a simple two-dimensional non-linear system and section 5 presents the final remarks and future directions for the work.

2 BACKGROUND

The systems considered in the paper are of the form,

$$\begin{cases} \dot{x}(t) = f(x(t),u(t)), & \text{for almost all } t \\ u(t) \in \mathcal{U}, \quad x(0) = x_0 \in \mathcal{X} \end{cases} \tag{1}$$

Where, $\mathcal{X}$, $\mathcal{U}$, are sets in a metric space standing, respectively, for a state space and a compact control space. Both are considered non-empty.

The conditions for existence and uniqueness of the solutions of the differential inclusion (1) are discussed below. Consider a set-valued map $F : \mathcal{X} \rightsquigarrow \mathcal{Y}$, with $\mathcal{Y}$ a set in a metric space, defined as,

$$\forall x \in \mathcal{X}, \quad F(x) := \{f(x,u)\}_{u\in\mathcal{U}} = \bigcup_{u\in\mathcal{U}} f(x,u). \tag{2}$$

This is a common extension of a wide class of differential equation models accounting for uncertainties (see for instance (Smirnov, 2002), pp. *xiv*). We can reformulate the system (1) as,

$$\dot{x}(t) \in F(x(t)), \qquad \text{for almost all } t > 0. \tag{3}$$

Continuity properties play a fundamental role in the existence of solutions of systems in the form (1). These are briefly reviewed below.

Definition 1. (Aubin and Cellina, 1984) F is *upper semicontinuous* (USC) at $\bar{x} \in X$ if for any open N containing $F(\bar{x})$ there exists a neighborhood M of $\bar{x}$ such that $F(M) \subset N$. F is a upper semicontinous set-valued map if it is USC for every $\bar{x} \in X$.

Definition 2. (Aubin and Cellina, 1984) F is *lower semicontinuous* (LSC) at $\bar{x} \in \mathcal{X}$ if for any $\bar{y} \in F(\bar{x})$ and any neighborhood $N(\bar{y})$, there exists a neighborhood $M(\bar{x})$ of $\bar{x}$ such that $\forall x \in M(\bar{x}), F(x) \cap N(\bar{y}) \neq \emptyset$

F is a lower semi-continuous set-valued map if it is LSC for every $\bar{x} \in X$.

The following result summarizes the continuity properties of the dynamics map that are key to the existence of solutions of (3).

Theorem 1. *Consider (2), with $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{U}$ subsets of metric spaces. If (i) $f : \mathcal{X} \times \mathcal{U} \mapsto \mathcal{Y}$ is C^1 function in both variables, (ii) for each $x \in \mathcal{X}$, f is convex relative to the second variable, u, and (iii) $\mathcal{U}$ is a non-empty, compact, convex set, then $\forall x \in \mathcal{X}, F(x)$ is a continuous (LSC and USC) and Lipschitz set-valued map, with non-empty, compact and convex images.*

Proof: The demonstration follows by showing, in sequence, non-emptiness, compactness, convexity, Lipschitziness[1] and each semi-continuity.

Non-emptiness:

Since $\mathcal{U} \neq \emptyset$ and $f(\cdot)$ is a continuous function in both variables, the set $F(x)$ has always at least one element, thus, it is non-empty.

Compactness:

The subset $\mathcal{U}$ being compact means that every sequence, u_n, in $\mathcal{U}$ has a convergent subsequence. By definition of uniform continuity, (see for instance (Ross, 2013)), with n a sucession index,

$$\forall (x,u_n)_{n\in\mathbb{N}} \subset \mathcal{X} \times \mathcal{U},$$
$$\lim_{n\to\infty}(x,u_n) = (x,\bar{u}) \implies \lim_{n\to\infty} f(x,u_n) = f(x,\bar{u}).$$

This means that, for every convergent subsequence in $\mathcal{U}$, we have a corresponding convergent subsequence $f(x,u_n) \in F(x)$. Therefore, every sequence of elements in $F(x)$ has at least one convergent subsequence, the set is then sequentially compact (by definition of compactness). Every sequentially compact metric space is compact [2], so, $F(x)$ is compact $\forall x \in \mathcal{X}$.

Convexity:

The convexity follows from the fact that $\forall x \in \mathcal{X}$, $f(x,u)$ is a convex, continuous function of $u \in \mathcal{U}$ and $\mathcal{U}$ a convex set. The range of such function, for each x, is a convex set.(Prop. 2.32 (Rockafellar and Wets, 1998)).

Lipschitz:

The set-valued map is globally Lipschitz at x if there exists a positive constant L such that,

$$\forall x_1, x_2 \in \mathcal{X},\ F(x_1) \subset F(x_2) + \mathcal{B}_2(0, L||x_1 - x_2||),$$

define, $d(f_1,f_2) := ||f(x_1,u_1) - f(x_2,u_2)||, \forall x_1,x_2 \in \mathcal{X}, \forall u_1,u_2 \in \mathcal{U}$, this implies that[3],

$$\begin{aligned} & d(f_1,f_2) \leq L||x_1 - x_2|| \Rightarrow \\ \Rightarrow\ & d(f_1,f_2)^2 \leq L^2||x_1 - x_2||^2 \\ & \leq L^2(||x_1 - x_2||^2 + ||u_1 - u_2||^2) \Rightarrow \\ & d(f_1,f_2) \leq L||(x_1,u_1) - (x_2,u_2)|| \end{aligned} \tag{4}$$

Since f is C^1, it is a Lipschitz function,

$$\forall x_1,x_2 \in \mathcal{X}, \forall u_1,u_2 \in \mathcal{U},$$
$$||f(x_1,u_1) - f(x_2,u_2)|| \leq M||(x_1,u_1) - (x_2,u_2)||, \tag{5}$$

[1] The property of being Lipschitz.

[2] Bolzano-Weierstrass Theorem.

[3] $||(x,y)||^2 = x_1^2 + ... + + x_p^2 + y_1^2 + ... + y_n^2 = ||x||^2 + ||y||^2$

the constant L, in (4) identifies with M in (5), so there exists a constant such that F is Lipschitz.

Upper Semi-Continuity:

Denote $\mathring{\mathcal{B}}(a,r)$ as the open ball of center a and radius r with some metric. To show USC, assume that $F()$ is not USC everywhere.

Therefore,

$$\exists \bar{x} \in \mathcal{X}, \exists N \supset F(\bar{x}), \forall M(\bar{x}), F(M(\bar{x})) \not\subset N \implies \implies \forall x \in M(\bar{x}), \forall u \in \mathcal{U}, f(x,u) \notin N \quad (6)$$

Knowing that x is in a neighborhood of $\bar{x}$,

$$\forall u \in \mathcal{U}, \exists \delta > 0, |(x,u) - (\bar{x},u)| < \delta \implies \implies \forall \varepsilon > 0, |f(x,u) - f(\bar{x},u)| < \varepsilon,$$

that is equivalent to,

$$\forall u \in \mathcal{U}, \forall \varepsilon_u > 0, f(x,u) \in \mathring{\mathcal{B}}(f(\bar{x},u), \varepsilon_u) \implies \implies \forall \varepsilon_u > 0, F(x) \subset \bigcup_{u \in \mathcal{U}} \mathring{\mathcal{B}}(f(\bar{x},u), \varepsilon_u). \quad (7)$$

To exist a neighborhood N, for $F(\bar{x})$ means that,

$$\exists \varepsilon > 0, \forall y \in F(\bar{x}), \mathring{\mathcal{B}}(y,\varepsilon) \subset N \Leftrightarrow \Leftrightarrow \exists \varepsilon > 0, \bigcup_{u \in \mathcal{U}} \mathring{\mathcal{B}}(f(\bar{x},u), \varepsilon) \subset N, \quad (8)$$

by the transitive property of inclusions using (7) and (8),$\forall x \in M(\bar{x}), F(x) \subset N$

This contradicts the proposition in (6), therefore, F must be USC.

Lower Semi-Continuity:

By one hand, admitting that F is not LSC everywhere implies,

$$\exists \bar{x} \in \mathcal{X}, \exists \bar{y} \in F(\bar{x}), \exists N(\bar{y}), \forall M(\bar{x}), \exists x \in M(\bar{x}), F(x) \cap N(\bar{y}) = \emptyset \implies \forall u \in \mathcal{U}, f(x,u) \notin N(\bar{y}) \implies \nexists \varepsilon > 0, |f(x,u) - \bar{y}| < \varepsilon. \quad (9)$$

By the other, using the definition of continuity,

$$\forall \bar{x} \in \mathcal{X}, \forall x \in M(\bar{x}), \forall u \in \mathcal{U}, \exists \rho > 0, |(x,u) - (\bar{x},u)| < \rho \implies \implies \forall \varepsilon > 0, |f(x,u) - f(\bar{x},u)| < \varepsilon,$$

and, as it is valid for any u and, $\bar{y} = f(\bar{x},u)$,

$$\forall \bar{y} \in F(\bar{x}), \exists \varepsilon > 0, |f(x,u) - \bar{y}| < \varepsilon,$$

this contradicts (9), therefore, F is also lower semi-continuous.

□

The following set operations will be of great use throughout the paper,

Definition 3. The Minkowski Sum and Pontryagrin Difference between two sets are defined, respectively, as,

$$A \oplus B := \{a + b \mid \in A, b \in B\},$$
$$A \ominus B := \{a \mid a \oplus B \subseteq A\}.$$

A ball is a set in some metric space $\mathcal{M}$, with some norm p, defined as,

$$\mathcal{B}_p(x,r) := \{y \in \mathcal{M} \mid ||y - x||_p \leq r\}.$$

2.1 Reachability and Viability

For the proper functioning of the algorithm proposed in this work (Section 3), we have to ensure that the reachable set can be computed and to properly define viability kernel. Definitions of both sets are laid out in this subsection. Afterwards we will explore the conditions of existence of general finite-time reachable sets, independently of the method of discretization or linearization.

Definition 4. The finite time horizon, $T = N_\rho \rho$, reachable set of the continuous non-linear system (1) with piecewise constant input, with initial states in the set $\mathcal{K} \subseteq \mathcal{X}$, is defined as,

$$\text{Reach}_T(\mathcal{K}) := \{x \in \mathcal{X} \mid x = x(T), x_0 \in \mathcal{K} : \exists \{u_0, ..., u_{N_\rho - 1}\} \in \mathcal{U}\} \quad (10)$$

Proposition 1. *Let a function $f : \mathcal{X} \times \mathcal{U} \mapsto \mathcal{Y}$ be defined as in* (2)*, and such that the conditions of Theorem 1 hold. Then, the finite-horizon reachable set,* (10) *can always be computed, for any $T \in \mathbb{R}^+$.*

Proof: From Theorem 1, $F(x)$ is non-empty, with compact, convex values and LSC, therefore we are in the conditions of Theorem 1 of (Aubin and Cellina, 1984) (pp. 97). This implies the existence of at least one continuously differentiable function, $x(t)$ for some tight interval. Since the properties of $F(x)$ are verified everywhere in its domain, the local existence of solutions becomes global. Therefore, $\forall T \in \mathbb{R}^+$, there exists $x(t)$, $\forall t \in [0,T]$.

□

We will also properly define the main object in the scope of this work,

Definition 5. The finite-horizon sampled-data viability kernel of $\mathcal{K}$, is the set of all the initial states in $\mathcal{K}$ for which there exists at least one trajectory $x_k := x(k\rho)$, with piecewise-constant input, that stays inside the constraint set $\mathcal{K}$ for the time horizon $T = \rho N_\rho$.

$$\text{Viab}_T^{sd}(\mathcal{K}) := \{x_0 \in \mathcal{K} \mid \forall i \in \{0, ..., N_\rho - 1\}, \exists u_i \in \mathcal{U} : x(t) \in \mathcal{K}, \forall t \in [0,T]\}.$$

2.2 Discretization of Linear Systems

A linear system is described by the following differential equation,

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + l(t), \quad \forall t > 0 \\ x(t) \in \mathcal{X}, u(t) \in \mathcal{U}, l(t) \in \mathcal{L}, x(0) = x_0. \end{cases} \tag{11}$$

With $\mathcal{X} \subset \mathbb{R}^n$, $\mathcal{U} \subset \mathbb{R}^d$ and $\mathcal{L} \subset \mathbb{R}^n$ compact, convex subsets. In the system above, $l(t)$ is a perturbation of the system, in our case, the linearization error. Define the sampling time as ρ. The time horizon considered is denoted as $\bar{T}$, with the number of time steps being, $N_\rho = \lfloor \bar{T}/\rho \rfloor$. Define the integral, $I_\rho := \int_0^\rho e^{Ar} dr$. The sampled-time discrete system becomes,

$$x_{k+1} = A_\rho x_k + B_\rho u_k + I_\rho l_k, \quad \forall k \in \{0, ..., N_\rho - 1\}, \tag{12}$$

with, $A_\rho := e^{A\rho}$, $B_\rho := I_\rho B$.

The state of the system at some time step s, with some initial state x_0, is given by,

$$x_s = A_\rho^s x_0 + \sum_{i=0}^{s-1} A_\rho^i (B_\rho u_{s-i-1} + I_\rho l_{s-i-1}) \tag{13}$$

2.3 Linearization

The linearization of the system (1) is done using some results in (Althoff, 2010), in the following manner,

$$\dot{x} \in f(x^*, u^*) + A(x - x^*) + B(u - u^*) + \mathcal{L}. \tag{14}$$

Matrices A and B are the Jacobian matrices of the respective linearizations, in terms of state and input variables, respectively, around the point (x^*, u^*). The term $\mathcal{L}$ is the Lagrangian Remainder. Let i be the index of the i-th state variable, defining $z = (x, u)$, the Hessian of the i-th state variable is,

$$J_i(\xi) := \frac{\partial^2 f_i(\xi)}{\partial z^2},$$

where, the i-th component of the remainder, $\mathcal{L}_i = \frac{1}{2}(z - z^*)^T J_i(\xi)(z - z^*)$, with, $\xi \in \{z^* + \alpha(z - z^*) \mid \alpha \in [0, 1]\}$.

This is the continuous time, linearized system. This system must be converted to a discrete-time version, since that is the representation in the scope of this work. Combining(14) with (12),

$$\begin{cases} \hat{x}_{k+1} = A_\rho^{(k)} \hat{x}_k + B_\rho^{(k)} u_k + \\ \qquad + I_\rho(z^*)(f(z^*) - A^{(k)} x_k^* - B^{(k)} u_k^*) \\ \hat{x}_k \in \mathcal{X}, u_k \in \mathcal{U}, z^* \in \mathcal{X} \times \mathcal{U}, \forall k \in \mathbb{N}^0 \end{cases} \tag{15}$$

The step reachable set of such linearized and discretized system (15), from the set $\mathcal{E}$, around the point z^* is computed as in (Girard et al., 2006),

$$\begin{aligned} \text{Reach}_\rho^{lin}(\mathcal{E}) := A_\rho[\mathcal{E}] \oplus B_\rho[\mathcal{U}] \\ + I_\rho(f(z^*) - Ax^* - Bu^*) \end{aligned} \tag{16}$$

Usually, it is hard to compute the exact set generated by the Lagrangian remainder. For that matter, an over-approximation is required for it (Althoff, 2010).

With the above formula (16), we can always compute the reachable set if we are in the following conditions:

Proposition 2. *Given a function $f : \mathcal{X} \times \mathcal{U} \mapsto \mathcal{Y}$ defined as in* (2)*. If f is once differentiable in both variables in all its domain, then, for any $x_k \in \mathcal{X}, u_k \in \mathcal{U}, (x^*, u^*) \in \mathcal{X} \times \mathcal{U}$, the step reachable set* (16) *can be computed.*

Proof: Since f is everywhere differentiable, the matrices A and B (defined in (14)) always exist. It is known that the power series defining e^{tA}, converges uniformly on compact time intervals, then A_ρ can be computed. The integral $\int_0^\rho e^{\lambda A} d\lambda$ also converges, so B_ρ and I_ρ exist and can be computed. Since differentiability implies continuity, $f(x^*, u^*)$ also exists. Therefore, the step reachable set can be computed always.
□

2.3.1 Polytopes

A convex $\mathcal{V}$-polytope, represented by a finite set of points, $\{v_1, v_2, ..., v_k\}$, is the convex hull of such set of points. $P = \text{Conv}(\{v_1, v_2, ..., v_k\})$.

If we stack every linear inequality that defines the polytope we have,$P = \{x \in \mathcal{X} \mid Hx \leq b\}$, where the stacked matrix H and the vector b represent the intersection of all the half spaces (linear inequalities) that define the polytope.

2.3.2 Zonotopes

A special class of convex polytopes are the *zonotopes*. They will be the main type of representation for reachable sets in this work. In (Girard, 2005) this representation is fully explained.

Definition 6. (Zonotope)

A zonotope is a set defined in an Euclidean space $\mathbb{R}^n$, $Z := \left\{x \in \mathbb{R}^n \mid x = c + \sum_{i=1}^p \alpha_i g_i, \ -1 \leq \alpha_i \leq 1\right\}$, where, c is the center of the zonotope, and, $g_1, ..., g_p$ are its generators, all of them are vectors in $\mathbb{R}^n$. Denote the zonotope, Z, as the tuple, $Z := (c, < g_1, ..., g_p >)$.

3 AN ALGORITHM FOR APPROXIMATING VIABILITY KERNELS

The main part of the algorithm overlaps with the work done in (Gillula et al., 2014). However, our algorithm is the result of modifying the feasibility assessment of trajectories, which results in a faster procedure and enabling it to be used for non-linear systems.

3.1 Basic Algorithm

The original structure of the algorithm, Algorithm 1, is based on sampling a number of random line segments, N_r, starting from a given point v_0 in the viability kernel and ending on the intersection with the boundary of the constraint set $\mathcal{K}$. By means of bisecting that line segment, the algorithm extracts, up to some tolerance, the closest point to the border of the viability kernel, belonging to it. The under approximation is the convex hull, $\text{Conv}(\cdot)$, of the points that are guaranteed to be in the viability kernel.

Algorithm 1: (Gillula et al., 2014) Computes a polytopic under-approximation of $\text{Viab}_T^{sd}(\mathcal{K})$.

```
function POLYTOPIC-APPROX(K, v_0, N_r)
    V ← {v_0}
    for i from 1 to N_r do
        r ← SampleRay(v_0)
        b ← IntersectionOnBoundary(K, r)
        v_i ← BisectionFeasibilitySearch(K, b, K)
        V ← V ∪ {v_i}
    end for
    return Conv(V)
end function
```

If $F(x)$ is a non-empty, compact, convex set-valued map and x taking values in a compact convex set $\mathcal{X}$, we assure that the output of Algorithm 1 still works for this system and it is in fact an under-approximation of the viability kernel of the non-linear system.

Definition 7. (Kuratowski Set Convergence (P. L. Papini, 2015))

Given a sequence of non-empty subsets, $\{A_n\}_{n=1}^{\infty}$, of a Banach Space X, it converges to A in the sense of *Kuratowski*, with $d(x,A) := \inf_{y\in A}||x-y||$, if,

$$\liminf_{n\to\infty} A_n = \limsup_{n\to\infty} A_n = A,$$

where,

$$\limsup_{n\to\infty} A_n = \{x \in X \mid \liminf_{n\to\infty} d(x,A_n) = 0\}, \quad (17)$$

$$\liminf_{n\to\infty} A_n = \{x \in X \mid \limsup_{n\to\infty} d(x,A_n) = 0\}. \quad (18)$$

Theorem 2. *Consider the system defined by a differential equation such as* (2)*. Admitting that the state space, $\mathcal{X}$, constraint set, $\mathcal{K}$, and the input set, $\mathcal{U}$, are non-empty, compact convex sets, and $f : \mathcal{X}\times\mathcal{U} \mapsto \mathcal{Y}$ is a C^1 function, convex in the second variable for each fixed x. Given a number of random directions, N_r, and an initial point $v_0 \in \text{Viab}_T^{sd}(\mathcal{K})$, and let, $\mathcal{S} = \text{Polytopic-Approx}(\mathcal{K}, v_0, N_r)$, then,*

$$\mathcal{S} \subseteq \text{Viab}_T^{sd}(\mathcal{K}).$$

Proof: Recall that, $\mathcal{S} = \text{Conv}(v_0, v_1, ..., v_{N_r})$. We want to verify that,

$$v_i \in \text{Viab}_T^{sd}(\mathcal{K}), \quad i \in \{0,1,...,N_r\} \implies \quad (19)$$

$$\implies \text{Conv}(v_0, v_1, ..., v_{N_r}) \subseteq \text{Viab}_T^{sd}(\mathcal{K}). \quad (20)$$

A sufficient condition for this to hold is the viability kernel to be a convex set. Since we are in the conditions of Theorem 1, we can describe the system by a differential inclusion $\dot{x}(t) \in F(x(t))$, where $F(x)$ has the property (among others) of having non-empty, compact, convex values and being Lipschitz. As in (Saint-Pierre, 1994) and by the definition of finite time horizon viability kernel of the sampled data system (Def. 5), the finite-time viability kernel for the sampled-data system can be built recursively as[4],

$$\begin{cases} K_0 = \mathcal{K} \\ K_{n+1} = \{x \in K_n \mid K_n \cap \text{Reach}_\rho(x) \neq \emptyset\}. \end{cases}$$

Where, K_n is the the n step viability kernel. Using a proof from (Maidens et al., 2013),

$$x \in K_{n+1} \Leftrightarrow x \in K_n \wedge (\text{Reach}_\rho(x) \cap K_n \neq \emptyset)$$
$$\Leftrightarrow x \in K_n \cap \text{Back}_\rho(K_n)$$

Notice that the set $\text{Back}_\rho(K_n)$ is the set of initial states in which there are trajectories that end in K_n the next instant, this is the backwards reachability set. This set coincides with the reachable set of the system similar to (1) but with, $\dot{x}(t) \in -F(x(t))$, $x_0 \in K_n$, instead(Aubin et al., 2011)(Raczynski, 2011).

If the step reachable set of the inverse dynamics is convex, then, by construction, $\text{Viab}_T^{sd}(\mathcal{K}) = K_{N_\rho}$ is a convex set.

Define $G(x) := -F(x)$. Since it was just applied a linear map to the set-valued function, $G(x)$ is still non-empty, compact, convex valued, Lipschitz in $\mathcal{X}$. From (Wolenski, 1990), the step reachable set can be computed as[5],

$$\text{Reach}_\rho(x) = \lim_{N\to\infty} \left(I + \frac{\rho}{N} G\right)^N (x),$$

[4]For upper semi-continuous set-valued maps, which is the case here.

[5]If a set-valued map $G(x)$ is composed with itself N many times, we denote the resulting set-valued map as $G^N(x)$.

notice that, $\forall x \in \mathcal{K}, \forall N \in \mathbb{N}$, the set $\left(I + \frac{\rho}{N} G\right)^N (x)$ is convex, because the set valued map $S(x) = I + \frac{\rho}{N} G(x)$ is convex (linear maps and adding 1 to every element in every dimension of $G(x)$) and the composition of convex imaged set-valued maps on convex-imaged set-valued maps yield convex sets.

From (P. L. Papini, 2015), the limit of a sequence of convex sets is convex, therefore, K_n is convex $\forall n \in \mathbb{N}$, and therefore, $\text{Viab}_T^{sd}(\mathcal{K})$ is also convex. That means that $\mathcal{S} \subseteq \text{Viab}_T^{sd}(\mathcal{K})$. □

3.2 Modified Algorithm

The procedure adapted for a more general case of non-linear systems is how the feasibility of some point x_0 is determined. The initial condition x_0 of the system is *feasible* in some set if at least one trajectory starting from x_0 stays inside that set. Determining if a point is feasible in a non-linear system framework is done with Algorithm 2.

Algorithm 2: Determines if the evolution of the system with initial set X_0 is feasible at each iteration k in the set $\mathcal{K}_k^\flat$, using reachability.

```
function FEASIBLE(N_ρ, ρ, S, X_0, {K_1^♭, ..., K_{N_ρ}^♭})
    for k from 1 to N_ρ do
        X_k ← Reach_S(ρ[k−1, k], X_{k−1})
        if not Overlaps(X_k, K_k^♭) then
            return False
        end if
    end for
    return True
end function
```

Proposition 3. *Consider a point $x_0 \in \mathcal{X}$, if* FEASIBLE$(N_\rho, \rho, \mathcal{S}, x_0, \{\mathcal{K}_1^\flat, ..., \mathcal{K}_{N_\rho}^\flat\})$ *returns* True*, then $x_0 \in \text{Viab}_T^{\text{sd}}(\mathcal{K})$, with $T = \rho N_\rho$. Where $\mathcal{K}_k^\flat$ is the set obtained by applying Lemma 1 and 2 from (Gillula et al., 2014) to the set $\mathcal{K}$ at the instant k.*

Proof: From the algorithm returning True we have,

$$\forall k \in \{1, ..., N_\rho\}, X_k \cap \mathcal{K}_k^\flat \neq \emptyset \Rightarrow$$
$$\forall s \in \{0, ..., N_\rho - 1\}, \exists u_s \in \mathcal{U} : \hat{x}_{s+1} \in \mathcal{K}_{s+1}^\flat. \quad (21)$$

Using Lemma 2 from (Gillula et al., 2014) and rearranging the quantifiers, (21) implies that, $\forall s \in \{1, ..., N_\rho\}, \exists \{u_0, ..., u_{N_\rho - 1}\} : x_s \in \mathcal{K}^\flat$. Which, by Lemma 1 from (Gillula et al., 2014), $x_0 \in \text{Viab}_T^{sd}(\mathcal{K})$. □

3.3 Computing Reachable Sets of Non-linear Systems

Propositions 1 and 2, establish the condition in which it is possible to compute the reachable set. The assurance of these conditions is very useful if one wants to guarantee that the algorithm terminates with a definite answer and do not end with an ill-posed problem.

The reachable set computation using linearizations (Algorithm 3) receives as inputs the discretization step ρ, the system's non-linear model, $\mathcal{S}$, the input space, $\mathcal{U}$, the initial condition set of the system, X_0, and the admissible Lagrangian remainder $\bar{\mathcal{L}}$.

Algorithm 3: Computes the step reachability set of non-linear systems (Althoff, 2010).

```
function REACH(ρ, S, U, X_0, L̄)
    (A, B, L) ← Linearize(S, X_0)
    if L ⊈ L̄ then
        (X_0^(1), X_0^(2)) ← Split(X_0)
        R^(1) ← Reach(ρ, S, U, X_0^(1), L̄)
        R^(2) ← Reach(ρ, S, U, X_0^(2), L̄)
        return {R^(1), R^(2)}
    else
        R ← AX_0 ⊕ BU
        return {R}
    end if
end function
```

In algorithm 3 we need to split a zonotope in two sets, since the union of zonotopes is not necessarily a zonotope, one avoids doing so, and simply add them in a list, and checking each one.

The user must also provide the admissible Lagrangian remainder, $\bar{\mathcal{L}}$, before the splitting takes place. The admissible bound for the remainder is used in the erosion of the constraint set as in Lemma 1 from (Gillula et al., 2014). The splitting of zonotopes is done as in (Althoff, 2010),

3.4 Overlap between a Zonotope and a $\mathcal{H}$-polytope

In order to determine if there are trajectories that keep the system inside the constraint set, the test for overlap between the reachable set and the constraint set must be performed. Using the aforementioned definition of zonotope, one will pose this problem as a simple feasibility program. The program will consist in determining the existence of the coefficients α_i such that the zonotope generated respects the inequality constraints derived from the facet representation

of the constraint set. The inequalities from the constraint set are in the form of,

$$Ax \leq b. \tag{22}$$

Defining the matrix where the generators of the zonotope defining the reachable set are its columns and the vector of the corresponding coefficients, $G = \begin{bmatrix} g_1 & \cdots & g_p \end{bmatrix}$, $\alpha = \begin{bmatrix} \alpha_1 & \cdots & \alpha_p \end{bmatrix}^T$

Substituting x in (22) by the definition of zonotope, with column-vectors, yields,$A(c+G\alpha) \leq b$. This is equivalent to the standard notation of inequalities for optimization with α as the variable to optimize,$AG\alpha \leq b - Ac \Leftrightarrow A^*\alpha \leq b^*$.

Other restrictions have to be added since each α_i only assumes values in $[-1,1]$, so, the optimization problem becomes,

$$\begin{array}{ll} \min\limits_{\alpha} & 0 \\ \text{s.t.} & A^*\alpha \quad \leq b^*, -1 \leq \alpha_i \leq 1 \end{array} \tag{23}$$

The Overlaps$(\cdot)$ function, returning whether a zonotope Z and a polytope P intersect each other or not, is then defined as,

$$\text{Overlaps}(Z,P) = \begin{cases} \text{True}, & \exists\alpha \text{ in (23)} \\ \text{False}, & \text{o.w.} \end{cases}$$

4 RESULTS

This section presents simulation results that point to the correctness of the algorithm. For the sake of simplicity, we will restrict our experiments to two-dimensional systems. The system being tested is a simple pendulum on a cart, with the adequate dimensions, where x_1 is the pendulum angle from the upright position and x_2 its angular velocity. The system is described by the following non-linear differential equation.

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \sin x_1 - u\cos x_1 \end{cases} \tag{24}$$

Here $x = (x_1, x_2) \in \mathcal{X} \subset \mathbb{R}^2$ and $u \in \mathcal{U} = [-1,1]$. The goal here is to determine the set of admissible pairs of angular position and angular velocity for the pendulum be kept inside the constraint set, $\mathcal{K} = [-\pi/6, \pi/6] \times [-\pi/6, \pi/6]$, using the adapted algorithm for non-linear systems. It was set $\rho = 0.05$, the number of time steps, $N_d = 10$, the number of sampling rays was set to $N_r = 100$, the bisection accuracy $\varepsilon = 0.03$ and the linearization tolerance, $\max_{x \in \tilde{L}} |x| = 0.03$.

Figure 1 shows two viability kernels computed using two different methods. The one in lighter color is computed using Saint-Pierre's algorithm. As in

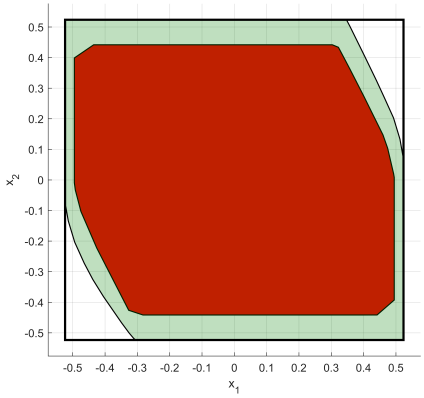

Figure 1: The outer line is the limit of the constraint set, the light colored (green) set is the finite horizon viability kernel with Saint-Pierre algorithm and the dark color (red) is the kernel computed with our adaptation.

(Saint-Pierre, 1994), this algorithm can also return the finite-time viability kernel of the system. It were used, $\rho = 0.05$, the space discretization step $h = 0.0033$, and the Lipschitz constant was set to $L = 2$, and it stopped at the $10th$ iteration so as to compute the 10-step viability kernel. The viable states computed by our algorithm are strictly contained in the one computed with the griding method. Since the accuracy of our method is user defined by the bisection accuracy, ε, and the acceptable Lagrangian remainder, $\overline{L}$, our solution can approximate the true $\text{Viab}_T^{sd}(\mathcal{K})$ as close as one would want, at the expense of more computations. Since the outer one is not constrained by having piecewise constant inputs, it always includes the one that is constrained.

5 CONCLUSIONS

The present work described an algorithm that adapts Gillula's *et al.* work on computing viability kernels for non-linear, sampled systems by linearizing and discretizing. We also prove the algorithm's correctness. The approach presented here reduces the finite horizon viability kernel computation problem to an iterative reachability one.

As noted in section 4, our algorithm under-approximates the true finite horizon viability kernel. In Figure 1 we can compare the infinite horizon viability kernel produced by Saint-Pierre's algorithm with ours. A system with piece-wise constant input will have less viable trajectories than one that can change inputs between sampling intervals (since it is a subset), so, $\text{Viab}_T^{sd}(K) \subseteq \text{Viab}_T(K)$, where the latter set is the finite-time viability kernel for the continuous time system (not sampled). The results confirm this

hypothesis, allowing us to confirm the correctness of the algorithm by simulation, for this case.

For a sufficiently large linearization error tolerance - does not require many set splitings - the proposed algorithm has a lower asymptotic complexity than that of strategies using state space griding, as in Saint-Pierre's type of methods, which is $O(a^n)$, for some constant a. In these cases our algorithm has a complexity similar to the one proposed in (Gillula et al., 2014), greater than $O(n^2)$ but smaller than $O(n^3)$.

There are two points to be careful when employing our algorithm. One is that methods similar to ours only yield finite horizon viability kernels. It is not possible to guarantee viable trajectories in infinite time, since for that the algorithm had to run forever. This is not a limitation from an application perspective. The other one is on the linearization errors. If one is largely tolerant on the linearization error, our under-approximation might be too conservative or even result in the empty-set due to the erosion process. If the tolerance is excessively strict, this may result in an enormous amount of splittings of sets, making the algorithm much slower.

Work on how to compute tight bounds for the Lagrangian Remainder is of great importance. The performance of the algorithm would be greatly improved if clever ways of splitting and storing the splitted sets were introduced.

Since the scope of this work is in the field of autonomous vehicles, this algorithm can be directly applied for sufficiently complex models. The computation of the viability kernels will be the basis in the synthesis of viable controls that keep the vehicle in safe states.

ACKNOWLEDGMENT

This work was partially supported by project FCT [UID/EEA/50009/2013].

REFERENCES

Althoff, M. (2010). *Reachability Analysis and its Application to the Safety Assessment of Autonomous Cars (PhD Thesis)*. PhD thesis.

Aubin, J., Bayen, A., and Saint-Pierre, P. (2011). *Viability Theory: New Directions*. Modern Birkhèauser classics. Springer Berlin Heidelberg.

Aubin, J. P. and Cellina, A. (1984). *Differential inclusions: set-valued maps and viability theory*. Springer-Verlag Berlin Heidelberg.

Gillula, J. H., Kaynama, S., and Tomlin, C. J. (2014). Sampling-based approximation of the viability kernel for high-dimensional linear sampled-data systems. *Proceedings of the 17th international conference on Hybrid systems: computation and control - HSCC 14*.

Girard, A. (2005). Reachability of uncertain linear systems using zonotopes. *Hybrid Systems: Computation and Control Lecture Notes in Computer Science*, page 291305.

Girard, A., Le Guernic, C., and Maler, O. (2006). Efficient computation of reachable sets of linear time-invariant systems with inputs. In Hespanha, J. P. and Tiwari, A., editors, *Hybrid Systems: Computation and Control*, pages 257–271, Berlin, Heidelberg. Springer Berlin Heidelberg.

Liniger, A. and Lygeros, J. (2017). Real-time control for autonomous racing based on viability theory. *IEEE Transactions on Control Systems Technology*, pages 1–15.

Maidens, J. N., Kaynama, S., Mitchell, I. M., Oishi, M. M., and Dumont, G. A. (2013). Lagrangian methods for approximating the viability kernel in high-dimensional systems. *Automatica*, 49(7):2017 – 2029.

P. L. Papini, S. W. (2015). Nested sequences of sets, balls, hausdor convergence. *Note di Matematica Volume 35, Issue 2*.

Raczynski, S. (2011). Uncertainty, dualism and inverse reachable sets. *International Journal of Simulation Modelling*, 10(1):38–45.

Rockafellar, R. T. and Wets, R. J.-B. (1998). *Variational analysis*. Springer.

Ross, K. A. (2013). *Elementary Analysis: The Theory of Calculus*.

Saint-Pierre, P. (1994). Approximation of the viability kernel. *Applied Mathematics & Optimization*, 29(2):187209.

Smirnov, G. V. (2002). *Introduction to the theory of differential inclusions*. American Mathematical Society.

Tiwary, H. R. (2008). On the hardness of computing intersection, union and minkowski sum of polytopes. *Discrete & Computational Geometry*, 40(3):469–479.

Wolenski, P. R. (1990). The exponential formula for the reachable set of a lipschitz differential inclusion. *SIAM Journal on Control and Optimization*, 28(5):1148–1161.