Nonlinear Analysis: Hybrid Systems 10 (2013) 45–62

Contents lists available at SciVerse ScienceDirect

Nonlinear Analysis: Hybrid Systems

journal homepage: www.elsevier.com/locate/nahs

Viable set computation for hybrid systems☆

Kostas Margellos *, John Lygeros

Automatic Control Laboratory, Department of Electrical Engineering, ETH Zürich, Physikstrasse 3, Zürich 8092, Switzerland

ARTICLE INFO

Keywords:
Hybrid systems
Viability
Optimal control
Differential game theory
Viscosity solutions
Lattice theory

ABSTRACT

In this paper, we revisit the problem of computing viability sets for hybrid systems with nonlinear continuous dynamics and competing inputs. As usual in the literature, an iterative algorithm, based on the alternating application of a continuous and a discrete operator, is employed. Different cases, depending on whether the continuous evolution and the number of discrete transitions are finite or infinite, are considered. A complete characterization of the reach-avoid computation (involved in the continuous time calculation) is provided based on dynamic programming. Moreover, for a certain class of automata, we show convergence of the iterative process by using a constructive version of Tarski's fixed point theorem, to determine the maximal fixed point of a monotone operator on a complete lattice of closed sets. The viability algorithm is applied to a benchmark example and to the problem of voltage stability for a single machine-load system in case of a line fault.

1. Introduction

1.1. Related work

The problem of synthesizing controllers for hybrid systems has attracted considerable attention both from the automatic control and the computer science community [1–4]. In this direction, [5–7] considered reachability and viability type of problems for hybrid automata. In [8,9] the authors characterize the maximal control invariant set (viability kernel in the sense of [10]) for a general class of hybrid systems, with nonlinear dynamics and competing inputs [11]. The proposed procedure was based on the alternating application of one continuous and two discrete operators. The former involves what was referred in [8] as reach-avoid computation, whereas the latter requires the inversion of the reset maps which encode the discrete behavior of the system.

There are certain limitations in the iterative procedure of [8]. The first is that there is no guarantee that the process reaches a fixed point, hence the algorithm might not converge to the desired viability kernel. Moreover, the continuous part of the algorithm involves a reach-avoid computation, which is hampered from a numerical point of view since the value function and the Hamiltonian of the optimal control problem involved are not necessarily continuous [8]. This is required so that the numerically computed solution (e.g. when using tools based on Level Set Methods) converges to the viscosity one as the numerical grid is made finer. The authors of [12,13], addressed this issue using the notion of viscosity solutions, but were restricted to the characterization of the "reach" part of the operator. To overcome these limitations and achieve a complete characterization of the algorithm, [14,15] considered the same problem from a viability theory perspective. Moreover, they introduced a theoretically sound notion of hybrid strategies in a gaming context and using nonsmooth analysis tools proved convergence of the iterative scheme.

☆ Research was supported by the European Commission under MoVeS, FP7-ICT-257005, the Network of Excellence HYCON2, FP7-ICT-257462, and the KIOS center of excellence 0308/28. This paper was presented in part at the IFAC Conference on Analysis and Design of Hybrid Systems [16].

* Corresponding author. Tel.: +41 762690133.
E-mail addresses: margellos@control.ee.ethz.ch (K. Margellos), lygeros@control.ee.ethz.ch (J. Lygeros).

1751-570X/$ – see front matter
http://dx.doi.org/10.1016/j.nahs.2013.06.001

1.2. Contributions

In this paper, we build on the approach introduced in [16] and provide an alternative framework for addressing viability problems for hybrid systems in an optimal control context, which supports theoretically the use of publicly available numerical tools [17] for hybrid reachability calculations. The proposed formulation is entirely based on optimal control and the properties of the hybrid system executions, and as such it serves as the optimal control counterpart of [14]. We consider different cases according to whether the time of continuous evolution and the number of discrete transitions are finite or infinite. We first restrict our attention to the case of finite discrete transitions and a finite time continuous evolution, as in [18]. For the finite time reach-avoid computation characterization we adopt the formulation of [19] which is based on the viscosity solution of a quasi-variational inequality of the form of [20,21]. Both the value function and the Hamiltonian of the optimal control problem are continuous, and hence the reach-avoid computation does not suffer from the drawbacks of [8,9]. We then consider the case where the continuous calculation is still of finite time, but an infinite number of transitions is allowed. To show convergence of the algorithm we reduce the problem to the calculation of the maximal fixed point of a monotone operator on a complete lattice [22,23], and based on transfinite recursion we use a constructive version of Tarski's theorem [24] to characterize this fixed point. Convergence of the algorithm may require limit ordinals higher than the first one, which limits the applicability of our approach to a certain class of hybrid automata (see discussion in Section 3.2), like those whose executions continue beyond the Zeno time. This approach, which is fundamentally different from the one adopted in [14], is related to the approach proposed by [25] for a more restricted class of hybrid dynamics. The more general case, deals with the problem of infinite time continuous evolution and infinite number of discrete transitions. In this case, we use the infinite time counterpart of the reach-avoid operator of [19], and follow a procedure similar to [20]. The last remaining case of infinite continuous evolution and finite number of discrete transitions follows directly from the results of Sections 3.1 and 3.3.

It should be noted that in [14] the authors prove similar results using techniques from nonsmooth analysis. Our derivation, not only serves as the optimal control counterpart of [14], but also exhibits certain differences with the work of [14]. Starting first from the case where we have finite time continuous evolution (this is the case in most applications), the proposed value function and the Hamiltonian of the optimal control problem are both continuous, thus enabling the use of existing numerical tools for viability computations (e.g. [17]) which are not supported by [14]. An additional difference with [14] appears in the situation where the number of discrete transitions is finite. In that case, which is not considered in [14], we provide a rigorous characterization of the set returned by the iterative algorithm (Proposition 2). Moreover, for the case where the number of discrete transitions is infinite we provide a convergence proof, which despite its limitations, is applicable to hybrid automata whose executions continue beyond the Zeno time. Such cases are not captured by [14].

We demonstrate some features of the proposed algorithms by means of a benchmark example and an application to the problem of voltage stability for a single machine-load system in case of a line fault [26,27]. The objective here is to determine the set of initial conditions for which the voltage will remain within the safety margins both during the transient phase and after the reclosure of the line.

The paper is organized as follows. Section 2 states the main assumptions, describes the hybrid dynamics, and poses the viability problem. Section 3 deals with the characterization of the continuous part of the proposed reachability algorithm, and the convergence of the iterative process for three different cases. Section 4 shows a benchmark example and the application of the viability algorithm to a power system case study. Finally, Section 5 summarizes our results and provides a list of open problems.

2. Viability specifications of hybrid game automata

2.1. Hybrid dynamics

We consider dynamical systems, whose state vector comprises both a discrete component q, and a continuous component x. The trajectories of the state vector are governed by control and disturbance inputs. Adopting the notation of [14], let υ and u denote the control, and δ, d the disturbance inputs (discrete and continuous respectively). The system can then be described by a hybrid automaton H.

Definition 1. A hybrid automaton H is the collection of

- discrete state variables $q \in Q$ and continuous state variables $x \in X$,
- control inputs $\upsilon \in V$ and $u \in U$,
- disturbance inputs $\delta \in \Delta$ and $d \in D$,
- vector field $f(\cdot, \cdot, \cdot, \cdot) : Q \times X \times U \times D \to X$,
- domain set, $\mathrm{Dom}(\cdot) : Q \to 2^X$,
- edges, $E \subseteq Q \times Q$,
- guard condition $G(\cdot) : E \to 2^X$,
- reset function $r(\cdot, \cdot, \cdot, \cdot) : E \times X \times V \times \Delta \to X$,

where 2^X denotes the power set of X. Throughout the paper we assume that $X = \mathbb{R}^n$. Before introducing the properties of the executions accepted by the hybrid automaton H, we provide the definition of a hybrid time set [28].

Definition 2. A hybrid time set $\tau = \{I_i\}_{i=0}^N$ is a finite or infinite sequence of intervals of the real line, such that for all $i < N$, $I_i = [\tau_i, \tau_i']$, if $N < \infty$, $I_N = [\tau_N, \tau_N')$ (possibly with $\tau_N' = \infty$), or $I_N = [\tau_N, \tau_N']$, and for all i, $\tau_i \leq \tau_i' = \tau_{i+1}$.

We are now in a position to define the class of executions accepted by the automaton H.

Definition 3. Let $\tau = \{I_i\}_{i=0}^N$ be a hybrid time set and consider the sequences of functions $\{q_i(\cdot)\}_{i=0}^N$, $\{x_i(\cdot)\}_{i=0}^N$, $\{\upsilon_i(\cdot)\}_{i=0}^N$, $\{u_i(\cdot)\}_{i=0}^N$, $\{\delta_i(\cdot)\}_{i=0}^N$, $\{d_i(\cdot)\}_{i=0}^N$, with $q_i(\cdot) : I_i \to Q$, $x_i(\cdot) : I_i \to X$, $\upsilon_i(\cdot) : I_i \to V$, $u_i(\cdot) : I_i \to U$, $\delta_i(\cdot) : I_i \to \Delta$, $d_i(\cdot) : I_i \to D$. The collection these sequences of functions is called execution of the hybrid automaton H starting from initial condition $(q_0(\tau_0), x_0(\tau_0))$, if and only if it satisfies the following conditions:

- Discrete evolution: For all $i < N$,
  1. $(q_i(\tau_i'), q_{i+1}(\tau_{i+1})) \in E$.
  2. $x_i(\tau_i') \in G(q_i(\tau_i'), q_{i+1}(\tau_{i+1}))$.
  3. $x_{i+1}(\tau_{i+1}) = r(q_i(\tau_i'), q_{i+1}(\tau_{i+1}), x_i(\tau_i'), \upsilon_{i+1}(\tau_{i+1}), \delta_{i+1}(\tau_{i+1}))$.
- Continuous evolution: For all i with $\tau_i < \tau_i'$
  1. $u_i(\cdot)$ and $d_i(\cdot)$ are Lebesgue measurable functions on I_i.
  2. $q_i(t) = q_i(\tau_i)$, $\upsilon_i(t) = \upsilon_i(\tau_i)$ and $\delta_i(t) = \delta_i(\tau_i)$ for all $t \in I_i$.
  3. $x_i(\cdot) : I_i \to X$ is the solution of the differential equation

  $$\dot{x}_i(t) = f(q_i(t), x_i(t), u_i(t), d_i(t)),$$

  over the interval I_i with initial condition $x_i(\tau_i)$.
  4. $x_i(t) \in \mathrm{Dom}(q_i(t))$ for all $t \in [\tau_i, \tau_i')$.

The executions accepted by H may be finite if τ is a finite sequence and its last interval is closed, infinite if τ is an infinite sequence or $\sum_{i=1}^{\infty}(\tau_i' - \tau_i) = \infty$, and Zeno if it is infinite but $\sum_{i=1}^{\infty}(\tau_i' - \tau_i) < \infty$. The convergence of the viability algorithm, and the validity of the results presented in the next section, rely on a series of assumptions on the hybrid automaton H and its executions.

Assumption 1. 1. The cardinality of Q is finite. The sets U, V, D and Δ are compact subsets of Euclidean spaces.
2. For all $q \in Q$ the function $f(q, x, u, d)$ is globally Lipschitz continuous in x, continuous in u and d and bounded. For all $(q, x) \in Q \times X$, the sets $\bigcup_{u \in U} f(q, x, u, d)$ and $\bigcup_{d \in D} f(q, x, u, d)$ are convex and compact for all $d \in D$ and all $u \in U$, respectively.
3. For all $q \in Q$, the set $\mathrm{Dom}(q)$ is open and $\mathrm{Dom}(q) \cup \bigcup_{q' \in Q} G(q, q') = X$.
4. For all $q, q' \in Q$ with $(q, q') \in E$, the function $r(q, q', x, \upsilon, \delta)$ is continuous in x, υ and δ.
5. For all $q, q' \in Q$ with $(q, q') \in E$, the set $G(q, q')$ is open, possibly the empty set.

The first two conditions in Assumption 1 are mainly imposed to ensure that in the gaming set-up certain sets are closed, whereas the convexity part is also used in Section 3.3. To ensure that f is bounded and since we are interested in the behavior inside a given set whose viability kernel (this set is denoted by F in Section 2.3) we seek to compute, we saturate f outside that set [12]. Note that f is automatically bounded in the numerical computations, since they are always performed on compact sets. Condition 3 guarantees that H is non-blocking [28]. Conditions 4 and 5 ensure continuity of the discrete operators of Section 2.3. Note that Zeno executions are not excluded from this formulation. Given this framework, and under Assumption 1, it was shown in [14] that for all admissible initial conditions and input trajectories there exists an infinite execution (possibly Zeno) for the automaton H.

2.2. Gaming formulation and input strategies

In purely continuous differential games [29,30], it is standard to consider the notion of nonanticipative strategies. Let $u(\cdot) \in \mathcal{U}$ and $d(\cdot) \in \mathcal{D}$, where $\mathcal{U}$, $\mathcal{D}$ denote the set of Lebesgue measurable functions $u(\cdot) : \mathbb{R}_+ \to U$ and $d(\cdot) : \mathbb{R}_+ \to D$, respectively. A function $\alpha(\cdot, \cdot, \cdot) : \mathcal{D} \times Q \times X \to \mathcal{U}$ is called non-anticipative (with respect to the first variable) if for all $(q, x) \in Q \times X$, $d(\cdot), d'(\cdot) \in \mathcal{D}$ and $T \geq 0$, $d(t) = d'(t)$ for almost every $t \in [0, T]$ implies that $\alpha(d, q, x)(t) = \alpha(d', q, x)(t)$ for almost every $t \in [0, T]$. Note that this definition is identical to the one of [31], with the difference that the nonanticipative strategy in this case depends also on (q, x), which will end up playing the role of the initial conditions of an interval of continuous evolution of the hybrid automaton. Under the choice of a nonanticipative strategy the player has the same information like feedback, but also knows the current value of the other player's input. This additional information compared to feedback strategies is needed to avoid situations where the game does not attain a value (see [32] for such a game).

Let $\mathcal{A}$ denote the class of nonanticipative strategies. Following [14], we can define a hybrid strategy (α, γ) for the control inputs (u, υ) as a pair whose first element is a nonanticipative strategy $\alpha(\cdot, \cdot, \cdot) : \mathcal{D} \times Q \times X \to \mathcal{U}$ for the continuous control input, and $\gamma(\cdot, \cdot) : Q \times X \to V$ is a feedback strategy for the discrete control input; note that in contrast to the continuous game a feedback strategy provides all necessary information to the discrete inputs. Note also that we do not allow transitions between discrete modes to be forced by neither the control nor the disturbance inputs (the domain and guard conditions do not depend on the controls or disturbances), and the discrete state after the transition is not determined by the control inputs. This restriction serves as a sufficient condition to ensure that the players are not second guessing one another in the discrete game. We could relax this assumption if the sets of states where the control and the disturbance inputs can force transitions are disjoint. However, autonomous transitions of the non-deterministic automaton are captured by the proposed framework.

2.3. Problem statement and definition of operators

The main objective is to characterize the hybrid discriminating kernel [14] of a given closed set $F \subseteq Q \times X$. This is equivalent to the maximal control invariant set of [9]. Formally this set is defined as follows (see Definitions 8 and 9 of [14] for details).

Let N denote a non-negative integer (to be thought of as the maximum number of allowable discrete transitions) and T denote a non-negative real number (to be thought of as the maximum allowable duration of continuous evolution), both possibly infinite. Given a hybrid strategy (α, γ), for $n \leq N$ and for disturbance actions $d(\cdot) \in \mathcal{D}$, $\{\delta_i\}_{i=0}^{n-1}$, the executions of the automaton satisfy Definition 3, and also for all $i < n$, and all $I_i \in \tau$ and $t \in I_i$, $\delta_i(t) = \delta_i$, $d_i(t) = d(t)$, $\upsilon_{i+1}(\tau_{i+1}) = \gamma(q_i(\tau_i'), x_i(\tau_i'))$, and $u_i(t) = \alpha(d_{\tau_i}(\cdot), q_i(\tau_i), x_i(\tau_i))(t - \tau_i)$, where $d_{\tau_i}(t) = d(t + \tau_i)$.

Definition 4. The hybrid discriminating kernel $\text{Viab}_F^{(N,T)}$ of a closed set $F \subseteq Q \times X$ is the set of $(q, x) \in F$, for which there exists a hybrid strategy (α, γ), such that for all $n \leq N$ and any disturbance actions $d(\cdot) \in \mathcal{D}$, $\{\delta_i\}_{i=0}^{n-1}$, all executions of the hybrid automaton starting from $(q_0(\tau_0), x_0(\tau_0)) = (q, x) \in F$ with $\sum_{i=0}^{n-1}(\tau_i' - \tau_i) \leq T$ and $\tau_n = \tau_n'$ are such that $(q_i(t), x_i(t)) \in F$ for all $I_i \in \tau$ and $t \in I_i$ with $i < n$, and $(q_n(\tau_n), x_n(\tau_n)) \in F$.

For technical purposes, in Definition 4 we consider executions that terminate with a transition. The algorithm developed below for computing such hybrid discriminating kernels is based on three set valued operators. Following [14], for an arbitrary set of states K, we define

$$\begin{aligned} \text{Pre}^{\exists}(K) = \{&(q, x) \in K \mid [x \notin \text{Dom}(q)] \wedge [\exists \upsilon \in V,\ \forall \delta \in \Delta, \\ &\forall q' \in Q,\ (q, q') \in E,\ x \in G(q, q') \Rightarrow (q', r(q, q', x, \upsilon, \delta)) \in K]\}, \end{aligned} \tag{1}$$

$$\begin{aligned} \text{Pre}^{\forall}(K) = K^c \cup \{&(q, x) \in K \mid \forall \upsilon \in V,\ \exists \delta \in \Delta, \exists q' \in Q, \\ &[(q, q') \in E, x \in G(q, q')] \wedge [(q', r(q, q', x, \upsilon, \delta)) \notin K]\}. \end{aligned} \tag{2}$$

The set $\text{Pre}^{\exists}(K)$ contains all states in K for which continuous evolution is not possible ($x \notin \text{Dom}(q)$), and there exists a choice for the discrete control input $\upsilon \in V$ such that for all disturbance inputs $\delta \in \Delta$, the state remains in K after one transition. On the other hand, $\text{Pre}^{\forall}(K)$, contains all states that are either outside K, or for all discrete control inputs $\upsilon \in V$ there exists a disturbance $\delta \in \Delta$ such that the state of the system leaves K after a transition.

For all $t \in [0, T]$[1] we define the continuous operator $\text{Reach}(t, \cdot, \cdot) : 2^{Q \times X} \times 2^{Q \times X} \to 2^{Q \times X}$, such that the set $\text{Reach}(t, R, A)$ includes all states $(q, x) \in Q \times X$ for which if the state starts at (q, x) at time t there exist a nonanticipative strategy α for the control inputs, such that for any disturbance d, the system trajectories either reach R via continuous evolution before passing through A, or remain outside A over the time interval $[t, T]$. In other words, $\text{Reach}(t, R, A)$ contains all states that are viable under continuous evolution, plus those that can reach R prior to hitting A. The latter was referred to as reach-avoid computation in [9]. Notice that the discrete state q remains constant along this computation, since only continuous dynamics are involved. Hence, for each $q \in Q$ we can define $R_q = \{x \in X \mid (q, x) \in R\}$ and $A_q = \{x \in X \mid (q, x) \in A\}$. Then, as in [14], treating one discrete state at a time,

$$\text{Reach}(t, R, A) = \bigcup_{q \in Q} \{q\} \times \text{Reach}_q(t, R_q, A_q), \tag{3}$$

where

$$\begin{aligned} \text{Reach}_q(t, R_q, A_q) = &\{x \in X \mid \exists \alpha(\cdot, q, x) \in \mathcal{A},\ \forall d(\cdot) \in \mathcal{D}, \\ &[\exists t_1 \in [t, T],\ \phi(t_1, t, q, x, \alpha(\cdot, q, x), d(\cdot)) \in R_q \\ &\wedge \forall t_2 \in [t, t_1],\ \phi(t_2, t, q, x, \alpha(\cdot, q, x), d(\cdot)) \in A_q^c \cap \text{Dom}(q)]\}, \\ &\cup \{x \in X \mid \exists \alpha(\cdot, q, x) \in \mathcal{A},\ \forall d(\cdot) \in \mathcal{D}, \\ &[\forall t_3 \in [t, T],\ \phi(t_3, t, q, x, \alpha(\cdot, q, x), d(\cdot)) \in A_q^c \cap \text{Dom}(q)]\}, \end{aligned} \tag{4}$$

where $\phi(\cdot, t, q, x, \alpha(\cdot, q, x), d(\cdot))$ denotes the solution of the continuous vector field at each discrete mode q starting from x at time t, when a nonanticipative strategy α is adopted. For each mode $q \in Q$, Eq. (4) implies that the set $\text{Reach}_q(t, R_q, A_q)$ is given by the union of two sets. The first contains all continuous states that can reach R_q at some time within our horizon, while avoiding passing through A_q until then. The second set includes all states that for the entire horizon remain outside A_q, without necessarily hitting R_q. We can compute $\text{Reach}_q(t, R_q, A_q)$ separately for each mode $q \in Q$, and use (3) to construct $\text{Reach}(t, R, A)$.

The operators defined above have the following basic properties, which are shown in [14] and are repeated here for completeness.

[1] This is to be understood as $[0, T)$ whenever $T = \infty$.

Algorithm 1 Finite time viability algorithm

1: **Initialization:**. $W_0 = F \times [0, T]$, $i = 0$.
2: **repeat**
$W_{i+1} = \text{Reach}(0, \text{Pre}^{\exists}(W_i), \text{Pre}^{\forall}(W_i))$,
$i = i + 1$.
3: **until** $W_i = W_{i-1}$ or $i = N$.
4: $\text{Viab}_{F\times[0,T]}^{(N,T)} = W_i$.
5: $\text{Viab}_{F}^{(N,T)} = \{(q, x) \in F \mid (q, x^z) \in \text{Viab}_{F\times[0,T]}^{(N,T)} \text{ and } z = 0\}$.

Proposition 1. *For all $t \in [0, T]$,*

1. *If R is closed and A is open, then R_q is closed, A_q is open and $\text{Reach}_q(t, R_q, A_q)$ is closed for all $q \in Q$.*
2. *For all $K \subseteq Q \times X$, $\text{Pre}^{\exists}(K) \subseteq \text{Reach}(t, \text{Pre}^{\exists}(K), \text{Pre}^{\forall}(K))$.*
3. *For all $K \subseteq Q \times X$, $\text{Reach}(t, \text{Pre}^{\exists}(K), \text{Pre}^{\forall}(K)) \subseteq K$.*
4. *The operator $\text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot))$ is monotone. For $K_1 \subseteq K_2 \subseteq Q \times X$, $\text{Reach}(t, \text{Pre}^{\exists}(K_1), \text{Pre}^{\forall}(K_1)) \subseteq \text{Reach}(t, \text{Pre}^{\exists}(K_2), \text{Pre}^{\forall}(K_2))$.*

The fact that $\text{Reach}_q(t, R_q, A_q)$ is closed if R_q is closed and A_q is open is established below, in Proposition 3 of Section 3.1. Note that if this holds for all $q \in Q$ then $\text{Reach}(t, R, A)$ is closed.

3. Hybrid discriminating kernel characterization

3.1. Case 1: finite continuous evolution and finite discrete transitions

Consider first the case where both T and N are finite (i.e. $T, N < \infty$). To keep track of the elapsed time we augment the hybrid automaton by introducing an additional continuous state and set $x^z = \begin{bmatrix} x & z \end{bmatrix} \in X \times \mathbb{R}$. For all $q \in Q$, $u \in U$, $d \in D$, the vector field of the augmented dynamics is given by $f^z(q, x^z, u, d) = \begin{bmatrix} f(q, x, u, d) & 1 \end{bmatrix} \in X \times \mathbb{R}$, and for all $q, q' \in Q$ with $(q, q') \in E$, $\upsilon \in V$, $\delta \in \Delta$ the reset map is given by $r^z(q, q', x, \upsilon, \delta) = \begin{bmatrix} r(q, q', x^z, \upsilon, \delta) & z \end{bmatrix} \in X \times \mathbb{R}$. Moreover, $\text{Dom}(\cdot) : Q \to 2^{X\times\mathbb{R}}$ and $G(\cdot) : E \to 2^{X\times\mathbb{R}}$. In words, the new state component $z \in \mathbb{R}$ counts forward in time and remains unaffected after discrete transitions.

The objective now becomes to compute the finite time hybrid discriminating kernel of a desired set $F \times [0, T] \subseteq Q \times X \times \mathbb{R}$. Given the discrete and continuous operators of Section 2.3, Algorithm 1 [14] summarizes the steps needed to compute $\text{Viab}_{F\times[0,T]}^{(N,T)}$. At iteration i of Algorithm 1 we compute the set W_{i+1} which includes the states from which executions can start and, despite any adversarial decision, can lead our system in $\text{Pre}^{\exists}(W_i)$ while avoiding passing through $\text{Pre}^{\forall}(W_i)$ until then. Based on the definition of the discrete operators, the previous statement implies that at iteration i we determine the states that remain in the predecessor set W_i following an execution comprising of one interval of continuous evolution followed by at most one discrete transition. The latter implies that the states in W_{i+1} remain viable (i.e. in W_0) following an execution with at most i discrete transitions. Therefore, the number of iterations of Algorithm 1 is related to the number of transitions that an admissible execution may contain.

Proposition 2. *The set $\text{Viab}_{F}^{(N,T)}$ computed by Algorithm 1 is indeed the desired finite time hybrid discriminating kernel of $F \in Q \times X$.*

The proof is given in the Appendix.

We now characterize $\text{Reach}_q(t, R_q, A_q)$ (for the augmented system) which based on (3) determines $\text{Reach}(t, R, A)$ for $t \in [0, T]$. This is achieved using tools from optimal control and viscosity solutions to Hamilton–Jacobi equations [31]. Notice that (4) is the union of two sets, hence we rewrite it as $\text{Reach}_q(t, R_q, A_q) = \widetilde{RA}(t, R_q, A_q) \cup N(t, A_q)$. $\widetilde{RA}(t, R_q, A_q)$ is a finite time reach-avoid computation in the sense of [19]. It includes all initial states from which trajectories can reach R_q at some time within $[t, T]$, without passing through A_q in the meantime. $N(t, A_q)$ is a finite time viability calculation in the sense of [12]. It contains all initial states from which trajectories stay in A_q^c during the interval $[t, T]$. Under the assumption that R_q is closed and A_q is open, consider two bounded, Lipschitz continuous functions $l(\cdot)$, $h(\cdot)$ such that

$$R_q = \{x^z \in X \times \mathbb{R} \mid l(x^z) \leq 0\},$$
$$A_q = \{x^z \in X \times \mathbb{R} \mid h(x^z) > 0\}.$$

Under these conditions, the set $\widetilde{RA}(t, R_q, A_q)$ was shown in [19] to be equal to the sub-zero level set of the function

$$\widetilde{V}(x^z, t) = \inf_{\alpha(\cdot)\in\mathcal{A}} \sup_{d(\cdot)\in\mathcal{D}} \min_{t_1\in[t,T]} \max\{l(\phi(t_1, t, x^z, \alpha(\cdot), d(\cdot))), \max_{t_2\in[t,t_1]} h(\phi(t_2, t, x^z, \alpha(\cdot), d(\cdot)))\}, \tag{5}$$

which is in turn the unique continuous viscosity solution (see [33] for a definition) of

$$\max\left\{h(x^z) - \widetilde{V}(x^z,t),\ \frac{\partial\widetilde{V}}{\partial t}(x^z,t) + \min\left\{0, \sup_{d\in D}\inf_{u\in U}\frac{\partial\widetilde{V}}{\partial x^z}(x^z,t)f(x^z,u,d)\right\}\right\} = 0, \tag{6}$$

over $(x^z,t) \in X \times \mathbb{R} \times [0,T]$, with $\widetilde{V}(x^z,T) = \max\{l(x^z), h(x^z)\}$.

Following [12], the set $N(t, A_q)$ is equal to the sub-zero level set of the function

$$\widehat{V}(x^z,t) = \inf_{\alpha(\cdot)\in\mathcal{A}}\sup_{d(\cdot)\in\mathcal{D}}\max_{t_3\in[t,T]} h(\phi(t_3,t,x^z,\alpha(\cdot),d(\cdot))), \tag{7}$$

which can be shown to be the unique continuous viscosity solution of

$$\frac{\partial\widehat{V}}{\partial t}(x^z,t) + \max\left\{0, \sup_{d\in D}\inf_{u\in U}\frac{\partial\widehat{V}}{\partial x^z}(x^z,t)f(x^z,u,d)\right\} = 0, \tag{8}$$

over $(x^z,t) \in X \times \mathbb{R} \times [0,T]$, with $\widehat{V}(x^z,T) = h(x^z)$.

We summarize our observations in the characterization of Reach in the following proposition.

Proposition 3. $\text{Reach}_q(t, R_q, A_q) = \{x^z \in X \times \mathbb{R} \mid \widetilde{V}(x^z,t) \le 0\} \cup \{x^z \in X \times \mathbb{R} \mid \widehat{V}(x^z,t) \le 0\}$.

Proof. Following Proposition 2 of [19], $\widetilde{RA}(t, R_q, A_q) = \{x^z \in X \times \mathbb{R} \mid \widetilde{V}(x^z,t) \le 0\}$ and $N(t, A_q) = \{x^z \in X \times \mathbb{R} \mid \widehat{V}(x^z,t) \le 0\}$. The statement follows then from the fact that $\text{Reach}_q(t, R_q, A_q) = \widetilde{RA}(t, R_q, A_q) \cup N(t, A_q)$. □

$\text{Reach}(t, R, A)$ can be then constructed from (3), taking the disjoint union of the sets $\text{Reach}_q(t, R_q, A_q)$. We note that unlike [9], the value functions $\widetilde{V}$ and $\widehat{V}$ are both continuous, which implies that the use of existing numerical tools [17,34] for approximating these functions is theoretically supported.

For the discrete part of the algorithm we need to compute the operators $\text{Pre}^{\exists}(\cdot)$ and $\text{Pre}^{\forall}(\cdot)$. This requires inversion of the reset map relations subject to existential and universal quantifiers, which is not an easy task in general. However, in the numerical examples of Section 4 the transition logic was simple enough to carry out this task by hand. Note also that since we treat every discrete mode separately in Algorithm 1, the computational complexity for each iteration is linear with respect to the number of discrete modes. However, this is not the case for the overall algorithm, apart from specific situations, like the case where the hybrid system can be represented by an acyclic graph.

For the continuous operator of Algorithm 1, and in particular to solve (6), (8), one could employ existing numerical tools based on Level Set Methods [17]. These are based on gridding the state space, therefore, the memory and computational cost grow exponential with the continuous dimension of the system. Assuming that an accurate enough grid is used, tight approximations of the (in general irregular and nonconvex) reachable sets can be achieved; however, this is not guaranteed to be an under-approximation or over-approximation. For a more detailed analysis and discussion on scalability issues the reader is referred to [35]. Note that (6), (8) are solved backwards in time; connections with forward reachability analysis can be found in [36].

3.2. Case 2: finite continuous evolution and infinite discrete transitions

In the case $T < \infty$, $N = \infty$ the theoretical computation of the hybrid discriminating kernel proceeds again via Algorithm 1, only without the terminating condition $i = N$. The computation of the set Reach required by the algorithm can be carried out as shown in Proposition 3. Unlike Section 3.1, infinite executions are also considered, however any such execution will be by definition Zeno. The iterations of the algorithm generate a sequence of nested closed sets whose intersection (possibly empty) is by construction a closed subset of $F \times [0,T]$. Consider then the complete lattice (see Section 3.2.2 for definitions) of closed subsets of $W_0 = F \times [0,T]$. By Proposition 1, $\text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot))$ is a monotone operator on this lattice, and hence by Knaster–Tarski theorem [22,23], it admits a maximal fixed point. Unfortunately, Tarski's theorem that could be used to characterize this fixed point is not constructive, unless the monotone operator is continuous. To overcome this difficulty, we adopt the constructive approach of [24], which relates the minimal and maximal fixed points of a monotone operator on a complete lattice to the limits of stationary transfinite iteration sequences. Following this approach no continuity assumption for the Reach operator is needed. We can thus relate the hybrid discriminating kernel to the fixed point of the set-valued operator Reach, but do not establish convergence of a countable intersection of sets to the operator's fixed point (the limit of the resulting transfinite iteration might not be $\bigcap_{i=0}^{\infty} W_i$), since sets corresponding to higher limit ordinals (possibly uncountable) may be required.

This limits the applicability of our theoretical analysis to many practically relevant cases, since most of the hybrid automata (including the Zeno ones) accept executions with a finite or infinite (but still countable) number of discrete transitions. Recall that the number of discrete transitions is related to the number of iterations in Algorithm 1. Therefore, convergence at a limit ordinal higher than the first one serves only as a correctness proof for such cases. However, there are

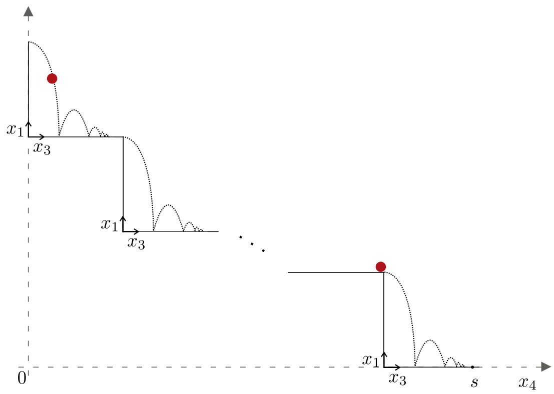

Fig. 1. Bouncing ball on a staircase.

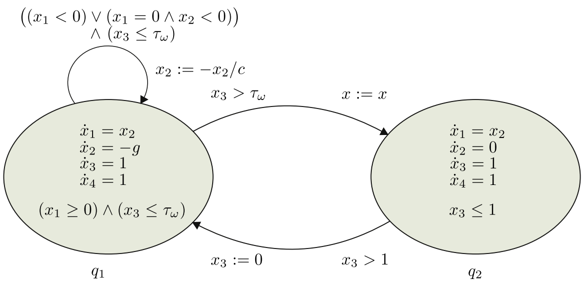

Fig. 2. Two-state hybrid automaton for the bouncing ball on a staircase example.

problems where the underlying automaton may accept executions whose number of transitions exceeds the first limit ordinal. This is for example the case when the objective is the continuation of the hybrid system executions beyond the Zeno time [37]. Convergence proofs like the one provided in [14] are not applicable to this class of systems, which are, however, captured by the proposed theoretical framework. For the rest of the section we focus on hybrid automata with a number of transitions higher than the first limit ordinal.

3.2.1. Motivating example

We provide here an example that better illustrates the class of hybrid systems where our theoretical framework is relevant. The so called bouncing ball problem is a typical example of a hybrid system with Zeno executions, and has been studied extensively in the hybrid system literature [38]. Here we consider a variant of the standard bouncing ball problem, where the ball is given a constant horizontal velocity and bounces on a staircase (see Fig. 1). At every stair the behavior of the ball reduces to the standard bouncing ball example. It bounces ω (the first limit ordinal) times, and after the Zeno time it rolls due to its horizontal velocity until it falls from the stair. The phenomenon is then repeated at all subsequent stairs.

The bouncing ball on a staircase problem can be modeled by the two-state hybrid automaton of Fig. 2. Assuming a local coordinate frame at each stair, x_1 denotes the vertical displacement of the ball at this stair, x_2 denotes its vertical velocity, x_3 is the horizontal displacement on the local frame and x_4 is a timer. Variable g denotes the gravitational acceleration and $c > 1$ is a constant. For simplicity we assumed $\dot{x}_3 = 1$ (unit horizontal velocity), therefore x_3 acts as a local timer as well, and after a transition to q_2 it is reset to zero. On the other hand, x_4 denotes the total time elapsed, which is equal in this case to the horizontal displacement of the ball with respect to the global coordinate frame. Mode q_1 corresponds to the standard bouncing ball problem. The ball stays in q_1 until the Zeno time $\tau_\omega = \frac{x_2(0)}{g} + \frac{(c+1)\sqrt{x_1^2(0)+2gx_2^2(0)}}{g(c-1)}$ (see also [38] for details), following an execution with ω transitions, however, at the same mode. Once x_3 exceeds τ_ω we have a transition at mode q_2, where the ball rolls until the end of the stair. We assumed that at the first stair $x_2(0) = 0$ so that the Zeno time is the same for all stairs, and also every stair is of unit length and height. When the length of the stair is exceeded we have a transition back to mode q_1, where the ball starts bouncing again. Here we assumed that the number of stairs is not finite; if we have a finite number of stairs an additional continuous state is required to keep track of the number of stairs elapsed.

We are interested in computing the hybrid discriminating kernel of $\{x \in \mathbb{R}^4 \mid x_4 \leq s\}$, i.e. the set of initial states from which the ball can start bouncing without exceeding s, where s is beyond the Zeno time of the last stair (see Fig. 1). By inspection of Fig. 1, the executions accepted by this automaton have a number of transitions higher than the first limit ordinal. In fact, at every stair we have ω transitions, and in the hypothetical set-up of an uncountable number of stairs the total number of transitions would also be uncountable. Applying the analysis of [14] for this problem is not possible. In the next subsections we show how this is captured by the proposed framework.

3.2.2. Tools from fixed point theory on lattices

We first recall the definition of a complete lattice [23], and provide the definition of an upper and lower iteration sequence and their limits [24]. Consider an arbitrary set K. An order (or ordering relation) on K is a binary relation $\leq$ such that for all $x, y, z \in K$, $x \leq x$, $x \leq y$ and $y \leq x$ imply $x = y$, $x \leq y$ and $y \leq z$ imply $x \leq z$; a set K with an ordering relation is called an ordered set. Note that the standard inclusion $\subseteq$ is an order for sets. An ordered set has a top element $\top$ if there exists $\top \in K$ such that $x \leq \top$ for all $x \in K$. The bottom element $\bot$ is defined in a similar way. For an arbitrary subset S of an ordered set K, if $\{x \in K | s \leq x \text{ for all } s \in S\}$ has a least element, then this element is called the supremum of S, denoted by $\vee S$. The infimum, $\wedge S$ of S is defined analogously.

Definition 5. A non-empty ordered set L is called a complete lattice if for all $S \subseteq L$, $\vee S$ and $\wedge S$ exist. Denote the complete lattice as $L(\leq, \top, \bot, \vee, \wedge)$.

Let now λ denote the smallest ordinal such that $\{i : i \in \lambda\}$ has cardinality strictly greater that the one of L. Ordinals could be thought of as a way to extend the natural numbers. Every ordinal denotes the set containing all strictly lower ordinals, but also the position of an element in a given sequence [39]. Moreover, since we consider lattices of sets, the ordering relation $\leq$ is to be understood as set inclusion.

Definition 6. The λ-termed lower (dually for the upper) iteration sequence for a monotone operator $P(\cdot) : L \to L$ starting with a set W_0 is the sequence $\langle W_i, i \in \lambda\rangle$ of elements of L, defined by the transfinite recursion

- $W_i = P(W_{i-1})$ for every successor ordinal $i \in \lambda$,
- $W_i = \bigcap_{j<i} W_j$ for every limit ordinal $i \in \lambda$.

Since we are considering a lower iteration sequence and $P(\cdot)$ is a monotone operator, it follows from Definition 6 that $W_i \leq W_{i-1}$.

Definition 7. A sequence $\langle W_i, i \in \lambda\rangle$ is stationary if and only if there exists $k \in \lambda$ such that for all $j \in \lambda$ with $j \geq k$, $W_j = W_k$. W_k is then the limit of the sequence. Denote by $\lim_P^l(W_0)$ ($\lim_P^u(W_0)$) the limit of a lower (upper) stationary sequence of a monotone operator P, starting with W_0.

Adopting the notation of [23], consider the sets of pre- and post-fixed points of P.[2]

$$\mathrm{Pre}(P) = \{x \in X \mid P(x) \leq x\}, \tag{9}$$

$$\mathrm{Post}(P) = \{x \in X \mid x \leq P(x)\}. \tag{10}$$

Following Tarski's theorem [23], the maximal and minimal fixed points of P are denoted by $\mathrm{gfp} = \vee\mathrm{Post}(P)$ and $\mathrm{lfp} = \wedge\mathrm{Pre}(P)$ respectively. To relate $\mathrm{gfp}(P)$ ($\mathrm{lfp}(P)$) to the limits of a lower (upper) iteration sequence, the following lemma is needed (dual to Lemma 3.1 and Theorem 3.2 (part 1) of [24]).

Lemma 1. *Let $\langle W_i, i \in \lambda\rangle$ be a λ-termed lower iteration sequence for the monotone operator $P(\cdot) : L \to L$, on the complete lattice L, starting with $W_0 \in L$. Then, if $\omega \in \lambda$ is the smallest limit ordinal,*

1. *For all $x \in L$ with $W_0 \geq x$ and $x \in \mathrm{Post}(P)$, we have that $W_i \geq x$ for all $i \in \lambda$.*
2. *For all $i \in \lambda$ let $a \leq i$ and $b < \omega$, such that $i = a\omega + b$. Then, for all $a' > a$ and for all $a'\omega \leq k \leq a'\omega + b$, $W_i \geq W_k$.*

The proof of this lemma is given in the Appendix.

The first part of Lemma 1 shows that for $W_0 \notin \mathrm{Post}(P)$, the lower iteration sequence $\langle W_i, i \in \lambda\rangle$ starting from W_0, can only reach $\mathrm{Post}(P)$ at some $k \in \lambda$ with $W_k = P(W_k) = W_{k+1}$ (i.e. stationarity). To see this apply part 1 of Lemma 1 with $x \in \mathrm{Post}(P)$ such that $x = P(x)$. The second part shows that for a lower iteration sequence $\langle W_i, i \in \lambda\rangle$, $\langle W_{i\omega}, i \in \lambda\rangle$ is also a decreasing chain. We are now in a position to show that if we start a lower iteration sequence from an initial set $W_0 \in \mathrm{Pre}(P)$, then a stationary decreasing chain is constructed, and its limit is the greatest fixed point of P less than or equal to W_0 [24, Theorem 3.2]. A pictorial representation is given in Fig. 3.

Lemma 2. *A λ-termed lower iteration sequence $\langle W_i, i \in \lambda\rangle$ for the monotone operator $P(\cdot) : L \to L$ on the complete lattice L starting with $W_0 \in \mathrm{Pre}(P)$, is a stationary decreasing chain, and its limit $\lim_P^l(W_0)$ is the greatest fixed point of P, less than or equal to W_0 (i.e. $\mathrm{gfp}(P) = \lim_P^l(W_0)$).*

The proof of this lemma is given in the Appendix.

3.2.3. Connection with the problem of hybrid discriminating kernel computation

We will now show how the fixed point theory of the previous subsection is related to the problem of computing hybrid discriminating kernels. Since we allow for executions with a number of transitions possibly higher than the first limit

[2] Note that in some references (e.g. [24]), $\mathrm{Pre}(P)$ is defined by (10) and $\mathrm{Post}(P)$ by (9) instead.

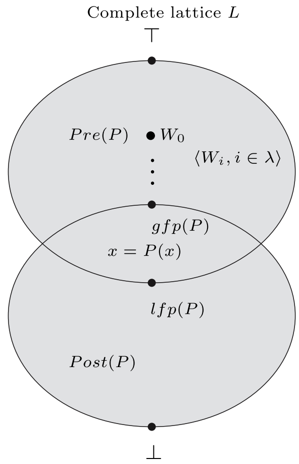

Fig. 3. Pictorial representation of the set of pre-fixed points Pre(P) and post-fixed points Post(P) of a monotone operator P on a complete lattice L. Starting from an initial set $W_0 \in \text{Pre}(P)$, a stationary, lower iteration sequence is constructed, which converges to the maximal fixed point gfp(P) of P.

ordinal, the definition of a hybrid time set and of an admissible execution (Definitions 2 and 3) should be revisited. To avoid repetitions and unnecessarily complicating our notation, we only state informally that the hybrid time set (similarly for the executions) should be thought of as a sequence of sets in the form of τ (see Definition 2), each of them having even an infinite number of intervals. The overall time set would then be a sequence of intervals whose number may correspond to higher limit ordinals.

Let now L be the complete lattice of closed subsets of $W_0 = F \times [0, T]$, and set

$$P = \text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot)).$$

Lemma 2 leads to the main result of this section.

Lemma 3. *There exists a limit ordinal $k \in \lambda$ such that for all $j \geq k$,*

$$\text{Reach}(t, \text{Pre}^{\exists}(W_j), \text{Pre}^{\forall}(W_j)) = W_k.$$

Moreover, W_k is the largest set such that this holds.

Proof. $\text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot))$ is a monotone operator on the complete lattice L. Therefore, by Knaster–Tarski theorem [23], it processes a maximal fixed point. Since $W_{i+1} \subseteq W_i$, $W_0 \in \text{Pre}(\text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot)))$. By Lemma 2, $\text{gfp}(\text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot))) = \lim^{l}_{W_0}(\text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot)))$. Therefore, by Definition 7 (and since for all $j \in \lambda$ we have that $W_{j+1} = \text{Reach}(t, \text{Pre}^{\exists}(W_j), \text{Pre}^{\forall}(W_j))$), there exists a limit ordinal $k \in \lambda$ such that for all $j \geq k$, we have $W_k = \text{Reach}(t, \text{Pre}^{\exists}(W_j), \text{Pre}^{\forall}(W_j))$. In Lemma 2 it was also shown that W_k is the greatest fixed point of $\text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot))$. Hence, W_k is the largest set such that this holds. □

Note that Lemma 3 guarantees the existence of a limit ordinal such that the viability algorithm converges, but not necessarily the least one. It remains to show that the fixed point W_k of the algorithm is the hybrid discriminating kernel of W_0. This can be done as in Theorem 2 of [14].

Theorem 1. *W_k is the hybrid discriminating kernel of $W_0 = F \times [0, T]$ (i.e. $W_k = \text{Viab}^{(N,T)}_{W_0}$ with N possibly infinite), with $k \in \lambda$ such that $W_k = \text{Reach}(t, \text{Pre}^{\exists}(W_k), \text{Pre}^{\forall}(W_k))$.*

Proof. The fact that $\text{Viab}^{(N,T)}_{W_0} \subseteq W_k$ can be shown as in the first part of Proposition 2. Following [14], to show that $W_k \subseteq \text{Viab}^{(N,T)}_{W_0}$ it suffices to show that W_k is hybrid discriminating domain.[3] By Theorem 1 of [14], this is equivalent with $W_k = \text{Reach}(t, \text{Pre}^{\exists}(W_k), \text{Pre}^{\forall}(W_k))$. The latter was shown in Lemma 3, and hence concludes the proof. □

Theorem 1 and Lemma 3 provide theoretical support when relating the fixed point of $\text{Reach}(t, \text{Pre}^{\exists}(\cdot), \text{Pre}^{\forall}(\cdot))$ to the hybrid discriminating kernel of W_0. However, the number of iterations required may exceed the first limit ordinal ω. Even though this is justified in hybrid automata like the one of Section 3.2.1 with more than ω transitions (e.g. continuation

[3] A closed set $K \subseteq Q \times X$ is called hybrid discriminating domain if there exists a hybrid strategy (α, γ), such that for all $n \leq N$ and any disturbance $d(\cdot) \in \mathcal{D}$, $\{\delta_i\}_{i=0}^{n-1}$, all executions of the hybrid automaton starting from $(q_0(\tau_0), x_0(\tau_0)) \in K$ with $\sum_{i=0}^{n-1} \tau'_i - \tau_i \leq T$ and $\tau_n = \tau'_n$ are such that $(q_i(t), x_i(t)) \in K$ for all $I_i \in \tau$ and all $t \in I_i$ with $i < n$, and $(q_n(\tau_n), x_n(\tau_n)) \in K$.

of executions beyond the Zeno time), algorithmic computation of the desired fixed point by means of Algorithm 1 is not possible. The reason is that we cannot repeat steps 2–3 of the algorithm more than ω times, and so the sets in Definition 6 that correspond to higher limit ordinals cannot be computed by Algorithm 1.

Revisiting now the bouncing ball on a staircase example of Section 3.2.1, it is clear that it falls in the framework of Theorem 1, for the specific set-up where no control or disturbance inputs are present. Clearly, the desired hybrid discriminating kernel of $\{x \in \mathbb{R}^4 \mid x_4 \le s\}$ (see Fig. 1) is the empty set. However, as discussed above, this cannot be computed by means of Algorithm 1 since the number of iterations cannot exceed ω; but this is the case for all other algorithms as well.

3.3. Case 3: infinite continuous evolution and infinite discrete transitions

This is the most general case, since both the number of discrete transitions and the horizon of the continuous evolution may be infinite (i.e. $T = \infty$, $N = \infty$). For the discrete part of the hybrid algorithm the same procedure as in the previous case can be followed. Therefore, the implications of Theorem 1 and the corresponding proofs remain unchanged, with the difference that the infinite time version of the continuous operator needs to be considered. To characterize this operator, define as in [13] the augmented input $\tilde{u} = (u, \bar{u}) \in U \times [0, 1]$, and consider the dynamics $\tilde{f}(x, \tilde{u}, d) = \bar{u}f(x, u, d)$. Note that since the horizon of the continuous evolution is allowed to be infinite, there is no need to augment the state space with an additional "timer" as in the previous subsections. Denote by $\widetilde{V}(x)$ and $\widehat{V}(x)$ the infinite horizon value functions, which correspond to (5) and (7) respectively. We will only provide the characterization of $\widetilde{V}(x)$; the same for $\widehat{V}(x)$. The infinite horizon value function $\widetilde{V}(x)$ is given by

$$\widetilde{V}(x) = \inf_{\alpha(\cdot)\in\mathcal{A}} \sup_{d(\cdot)\in\mathcal{D}} \max\{l(\tilde{\phi}(t, x, \alpha(\cdot), d(\cdot))), \max_{\tau\in[0,t]} h(\tilde{\phi}(\tau, x, \alpha(\cdot), d(\cdot)))\},$$

for all $t > 0$. The map $\tilde{\phi}(\cdot, x, \alpha(\cdot), d(\cdot)) : [0, \infty) \to X$ denotes the trajectory of the augmented system, starting from the initial condition x with inputs $\alpha(\cdot), d(\cdot)$. Unlike the finite time case of Section 3.1, there is no explicit dependency of $\tilde{\phi}$ on the initial time. Moreover, $\widetilde{V}(\cdot)$ is not necessarily continuous.

Lemma 4. *The function $\widetilde{V}(x)$ is upper semicontinuous.*

Proof. Consider an arbitrary $x_0 \in X$. It suffices to show that for all $\epsilon > 0$ there exists $\delta > 0$ such that

$$\widetilde{V}(x) - \widetilde{V}(x_0) < \epsilon \quad \text{for all } |x - x_0| < \delta.$$

By the definition of $\widetilde{V}(x)$, for all $\epsilon > 0$ there exists $T \in [0, \infty)$ and $\hat{\alpha}(\cdot) \in \mathcal{A}$ such that for all $d(\cdot) \in \mathcal{D}$,

$$\widetilde{V}(x_0) > \max\{l(\tilde{\phi}(T, x_0, \hat{\alpha}(\cdot), d(\cdot))), \max_{\tau\in[0,T]} h(\tilde{\phi}(\tau, x_0, \hat{\alpha}(\cdot), d(\cdot)))\} - \frac{\epsilon}{4}. \tag{11}$$

Due to the continuous dependence of finite time trajectories on initial conditions (recall that both $l(\cdot)$, $h(\cdot)$ are Lipschitz continuous), there exists $\delta > 0$ with $|x - x_0| < \delta$, such that for all $\alpha(\cdot) \in \mathcal{A}$ and $d(\cdot) \in \mathcal{D}$,

$$\left|\max_{\tau\in[0,T]} h(\tilde{\phi}(\tau, x, \alpha(\cdot), d(\cdot))) - \max_{\tau\in[0,T]} h(\tilde{\phi}(\tau, x_0, \alpha(\cdot), d(\cdot)))\right| < \frac{\epsilon}{4}, \tag{12}$$

$$\left|l(\tilde{\phi}(\tau, x, \alpha(\cdot), d(\cdot))) - l(\tilde{\phi}(\tau, x_0, \alpha(\cdot), d(\cdot)))\right| < \frac{\epsilon}{4}. \tag{13}$$

But, by the definition of $\widetilde{V}(x)$, we have that

$$\widetilde{V}(x) < \sup_{d(\cdot)\in\mathcal{D}} \max\{l(\tilde{\phi}(T, x, \hat{\alpha}(\cdot), d(\cdot))), \max_{\tau\in[0,T]} h(\tilde{\phi}(\tau, x, \hat{\alpha}(\cdot), d(\cdot)))\}.$$

Hence, there exists $\hat{d}(\cdot) \in \mathcal{D}$ such that

$$\widetilde{V}(x) < \max\{l(\tilde{\phi}(T, x, \hat{\alpha}(\cdot), \hat{d}(\cdot))), \max_{\tau\in[0,T]} h(\tilde{\phi}(\tau, x, \hat{\alpha}(\cdot), \hat{d}(\cdot)))\} + \frac{\epsilon}{2}.$$

Since (11)–(13) hold for any $d(\cdot) \in \mathcal{D}$, they would also hold for $\hat{d}(\cdot)$. We can then distinguish two cases. If $\widetilde{V}(x) < l(\tilde{\phi}(T, x_0, \hat{\alpha}(\cdot), \hat{d}(\cdot)))$, then statements (11), (13) lead to $\widetilde{V}(x) - \widetilde{V}(x_0) < \epsilon$. Else, if $\widetilde{V}(x) < \max_{\tau\in[0,T]} h(\tilde{\phi}(\tau, x_0, \hat{\alpha}(\cdot), \hat{d}(\cdot)))$, statements (11), (12) lead to $\widetilde{V}(x) - \widetilde{V}(x_0) < \epsilon$, and conclude the proof. □

Proposition 4. *The function $\widetilde{V}(x)$ is a viscosity solution of*

$$\max\left\{h(x) - \widetilde{V}(x), \min\left\{0, \sup_{d\in D} \inf_{u\in U} \frac{\partial \widetilde{V}}{\partial x}(x) f(x, u, d)\right\}\right\} = 0. \tag{14}$$

Note that (14) is the stationary version of (6) for the initial (not the augmented) system (see [19] for details). Following a proof similar to the finite horizon case (see Theorem 1 of [19]) it is straightforward to show that $\widetilde{V}(x)$ is a viscosity subsolution of (14). Under the convexity part of Assumption 1 we can also show that the lower semicontinuous envelope of $\widetilde{V}(x)$ is a viscosity supersolution of (14). This implies that $\widetilde{V}(x)$ will be a viscosity supersolution as well. Note that instead of imposing the second part of Assumption 1, one could enlarge the set of admissible control functions from the class of measurable functions to that of relaxed controls (see Chapter 3 of [31]). Unfortunately, the comparison principle does not hold in this case and hence the viscosity solution is not unique [20]. However, $\widetilde{V}(x)$ is the maximal upper semicontinuous viscosity subsolution.

Proposition 5. *For all upper semicontinuous viscosity subsolutions $W(x)$ of (14) such that $W(x) \leq l(x)$ for all $x \in X$, $\widetilde{V}(x) \geq W(x)$ for all $x \in X$.*

Proof. The proof is analogous to that of Proposition 5 of [20], and is based on the comparison principle for discontinuous viscosity solutions [31]. For any $T \in [0, \infty)$, let $w(x, t) = W(x)$ for all $t \in [0, T]$ and $x \in X$. We have $w(x, T) \leq l(x)$, and hence $w(x, T) \leq \max\{l(x), \widetilde{h}(x)\} = \widetilde{V}(x, T)$. Theorem 3.4 of [40] leads then to $\widetilde{V}(x, t) \geq w(x, t) = W(x)$ on $X \times [0, T]$. Therefore, $T \to \infty$ leads to $\widetilde{V}(x) = \lim_{T\to\infty} \widetilde{V}(x, T) \geq W(x)$. □

Computation of the desired hybrid discriminating kernel can be performed using Algorithm 1, without the terminating condition $i = N$ and with the difference that the continuous operator at step 2 of the algorithm is now replaced by its infinite horizon counterpart. The latter can be computed by means of Proposition 3, with the finite horizon value functions being replaced by $\widetilde{V}(x)$ and $\widehat{V}(x)$, respectively. Due to lack of continuity of the value functions (see Lemma 4) and since the numerical tools based on Level Set Methods [17] deal with finite horizon problems, to approximate $\widetilde{V}(x)$ (similarly for $\widehat{V}(x)$) numerically, instead of (14) we solve (6) until the solution does not evolve any more (i.e. the resulting sets have saturated). This procedure is also employed in the example of Section 4.2.

3.4. Case 4: infinite continuous evolution and finite discrete transitions

The computation of the hybrid discriminating kernel for the case where $T = \infty, N < \infty$ follows directly from the analysis of Sections 3.1 and 3.3. Specifically, Reach can be computed as shown in Section 3.3, whereas the hybrid discriminating kernel can be determined by Proposition 2, applying Algorithm 1 of Section 3.1. Even if convergence does not occur the algorithm will terminate after at most N iterations.

4. Case studies

4.1. Numerical example

Consider the hybrid automaton of Fig. 4 with $Q = \{q_1, q_2\}$, $X = \mathbb{R}$, $U = D = V = [-1, 1]$, $\Delta = [0, 1]$, $f(q_1, x, u, d) = 0$ and $f(q_2, x, u, d) = u + (\frac{1}{2} - x)d$. Notice also that $E = \{(q_1, q_2), (q_2, q_1)\}$, $\text{Dom}(q_1) = (-\infty, -\frac{1}{2}) \cup (\frac{1}{2}, \infty)$, $\text{Dom}(q_2) = (-\infty, 0)$, $G(q_1, q_2) = \mathbb{R}$, $G(q_2, q_1) = (-1, \infty)$, $r(q_1, q_2, x, v, \delta) = 2x\delta$ and $r(q_2, q_1, x, v, \delta) = x$. Clearly, the hybrid automaton satisfies all parts of Assumption 1.

We will now apply the infinite time counterpart of Algorithm 1 (i.e. $T = \infty$ and termination occurs only if $W_i = W_{i-1}$) with $W_0 = \{q_1\} \times [-1, 1] \cup \{q_2\} \times [-1, 1]$. Since we do not require the time of continuous evolution to be finite, the additional state that was appended to the continuous state vector to track time is no longer needed. For all $i = 0, 1, \ldots$, let $W_{i,q_1} = \{x \in \mathbb{R} | (q_1, x) \in W_i\}$, $W_{i,q_2} = \{x \in \mathbb{R} | (q_2, x) \in W_i\}$, and define

$$S_i(q_1) = \{x \in \mathbb{R} | \exists v \in V, \forall \delta \in \Delta, x \in G(q_1, q_2) \Rightarrow r(q_1, q_2, x, v, \delta) \in W_{i,q_2}\},$$
$$T_i(q_1) = \{x \in \mathbb{R} | \forall v \in V, \exists \delta \in \Delta, x \in G(q_1, q_2) \wedge r(q_1, q_2, x, v, \delta) \notin W_{i,q_2}\}.$$

$S_i(q_1)$ contains the states $x \in \mathbb{R}$ for which there exists a choice for the discrete control input $v \in V$ such that for all disturbance inputs $\delta \in \Delta$, the continuous state ends up in W_{i,q_2} after a transition, thus remaining in W_i. On the other hand, $T_i(q_1)$ contains all states $x \in \mathbb{R}$ for which for all discrete control inputs $v \in V$ there exists at least one choice for the disturbance $\delta \in \Delta$ such that the continuous state leaves W_i after a transition since it does not end up in W_{i,q_2}. The sets $S_i(q_2)$, $T_i(q_2)$ are defined analogously. Treating each mode $q \in Q$ separately, notice from (1), (2) that

$$\text{Pre}^{\exists}(W_{0,q_1}) = \text{Dom}^c(q_1) \cap S_0(q_1) = \left[-\frac{1}{2}, \frac{1}{2}\right] \cap \left[-\frac{1}{2}, \frac{1}{2}\right] = \left[-\frac{1}{2}, \frac{1}{2}\right],$$

$$\begin{aligned}\text{Pre}^{\forall}(W_{0,q_1}) &= W_{0,q_1}^c \cup T_0(q_1) = \big((-\infty, -1) \cup (1, \infty)\big) \cup \left(\left(-\infty, -\frac{1}{2}\right) \cup \left(\frac{1}{2}, \infty\right)\right) \\ &= \left(-\infty, -\frac{1}{2}\right) \cup \left(\frac{1}{2}, \infty\right),\end{aligned}$$

$$\text{Pre}^{\exists}(W_{0,q_2}) = \text{Dom}^c(q_2) \cap S_0(q_2) = [0, \infty) \cap (-1, 1] = [0, 1],$$

$$\begin{aligned}\text{Pre}^{\forall}(W_{0,q_2}) &= W_{0,q_2}^c \cup T_0(q_2) = \big((-\infty, -1) \cup (1, \infty)\big) \cup (1, \infty) \\ &= (-\infty, -1) \cup (1, \infty).\end{aligned}$$

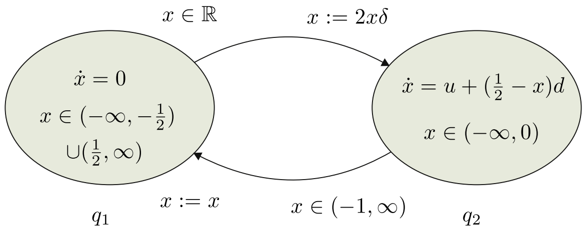

Fig. 4. Two-state hybrid automaton for the example of Section 3.2.

The fact that $S_0(q_1) = [-\frac{1}{2}, \frac{1}{2}]$ follows from the definition of S_0, where we seek to determine the set of $x \in \mathbb{R}$ such that $r(q_1, q_2, x, v, \delta) \in W_{i,q_2}$ for all $\delta \in \Delta$. The latter implies that for all $\delta \in [0, 1]$, $2x\delta \in [-1, 1]$. We can now compute $W_1 = \{q_1\} \times W_{1,q_1} \cup \{q_2\} \times W_{1,q_2}$, where

$$W_{1,q_1} = \text{Reach}_{q_1}(\text{Pre}^{\exists}(W_{0,q_1}), \text{Pre}^{\forall}(W_{0,q_1})) = \left[-\frac{1}{2}, \frac{1}{2}\right],$$

$$W_{1,q_2} = \text{Reach}_{q_2}(\text{Pre}^{\exists}(W_{0,q_2}), \text{Pre}^{\forall}(W_{0,q_2})) = \left[-\frac{1}{2}, 1\right].$$

For the computation of W_{1,q_2} it suffices to notice that for all $x \geq -1/2$ there exists u such that for all d, $f(q_2, x, u, d) \geq 0$. Proceeding on a similar way, the second iteration of the algorithm results in

$$\text{Pre}^{\exists}(W_{1,q_1}) = \text{Dom}^c(q_1) \cap S_1(q_1) = \left[-\frac{1}{2}, \frac{1}{2}\right] \cap \left[-\frac{1}{4}, \frac{1}{2}\right] = \left[-\frac{1}{4}, \frac{1}{2}\right],$$

$$\begin{aligned}\text{Pre}^{\forall}(W_{1,q_1}) &= W^c_{1,q_1} \cup T_1(q_1) = \left(\left(-\infty, -\frac{1}{2}\right) \cup \left(\frac{1}{2}, \infty\right)\right) \cup \left(\left(-\infty, -\frac{1}{4}\right) \cup \left(\frac{1}{2}, \infty\right)\right) \\ &= \left(-\infty, -\frac{1}{4}\right) \cup \left(\frac{1}{2}, \infty\right),\end{aligned}$$

$$\text{Pre}^{\exists}(W_{1,q_2}) = \text{Dom}^c(q_2) \cap S_1(q_2) = [0, \infty) \cap \left[-\frac{1}{2}, \frac{1}{2}\right] = \left[0, \frac{1}{2}\right],$$

$$\begin{aligned}\text{Pre}^{\forall}(W_{1,q_2}) &= W^c_{1,q_2} \cup T_1(q_2) = \left(\left(-\infty, -\frac{1}{2}\right) \cup (1, \infty)\right) \cup \left(\left(-1, -\frac{1}{2}\right) \cup \left(\frac{1}{2}, \infty\right)\right) \\ &= \left(-\infty, -\frac{1}{2}\right) \cup \left(\frac{1}{2}, \infty\right).\end{aligned}$$

Therefore, $W_2 = \{q_1\} \times W_{2,q_1} \cup \{q_2\} \times W_{2,q_2}$, where

$$W_{2,q_1} = \text{Reach}_{q_1}(\text{Pre}^{\exists}(W_{1,q_1}), \text{Pre}^{\forall}(W_{1,q_1})) = \left[-\frac{1}{4}, \frac{1}{2}\right],$$

$$W_{2,q_2} = \text{Reach}_{q_2}(\text{Pre}^{\exists}(W_{1,q_2}), \text{Pre}^{\forall}(W_{1,q_2})) = \left[-\frac{1}{2}, \frac{1}{2}\right].$$

In general, it is easy to show that for $i = 1, 2, \ldots$, $W_{2i-1} = \{q_1\} \times W_{2i-1,q_1} \cup \{q_2\} \times W_{2i-1,q_2}$ and $W_{2i} = \{q_1\} \times W_{2i,q_1} \cup \{q_2\} \times W_{2i,q_2}$, where

$$W_{2i-1,q_1} = \left[-\frac{1}{2^i}, \frac{1}{2^i}\right], \qquad W_{2i,q_1} = \left[-\frac{1}{2^{i+1}}, \frac{1}{2^i}\right],$$

$$W_{2i-1,q_2} = \left[-\frac{1}{2^i}, \frac{1}{2^{i-1}}\right], \qquad W_{2i,q_2} = \left[-\frac{1}{2^i}, \frac{1}{2^i}\right].$$

Hence, the sequence $\{W_i\}_i$ converges asymptotically ($N = \infty$) to $\text{Viab}^{(N,T)}_{W_0} = \{q_1\} \times \{0\} \cup \{q_2\} \times \{0\}$, which is the hybrid discriminating kernel of W_0. Notice that for all executions starting at $(q, 0)$ with $q \in Q$ the continuous state remains at zero either via continuous evolution or after a discrete transition. In particular, there exists the choice of a Zeno execution, taking at $\tau'_0 = \tau_0 = 0$ an infinite number of transitions between the discrete states q_1 and q_2, without time progressing further.

It should be also noted that, as stated in Propositions 2 and 4 of [14] and Proposition 1, the sets $\text{Reach}(\text{Pre}^{\exists}(W_i), \text{Pre}^{\forall}(W_i))$, $\text{Pre}^{\exists}(W_i)$ are closed and their volume decreases with the number of iterations (see Fig. 5), whereas $\text{Pre}^{\forall}(W_i)$ is open and its volume increases with the number of iterations. The latter is not included in Fig. 5 since it extends to infinity. Since $\text{Reach}(\text{Pre}^{\exists}(W_i), \text{Pre}^{\forall}(W_i))$ and $\text{Pre}^{\exists}(W_i)$ are closed intervals, their volume is defined as the length of the corresponding interval.

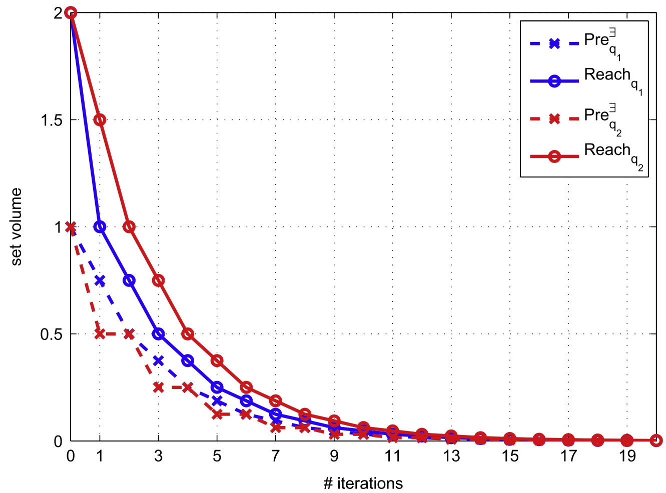

Fig. 5. Normalized volume of $\text{Pre}^{\exists}(W_{i,q})$ and $\text{Reach}_q(\text{Pre}^{\exists}(W_{i,q}), \text{Pre}^{\forall}(W_{i,q}))$ for every iteration $i = 0, 1, 2, \ldots$ and each mode $q \in Q$. Note that for all $i = 0, 1, 2, \ldots, \text{Pre}^{\exists}(W_{i,q_1}) = \text{Reach}_{q_1}(\text{Pre}^{\exists}(W_{i,q_1}), \text{Pre}^{\forall}(W_{i,q_1}))$.

4.2. Voltage stability of a single machine-load system

4.2.1. System description and mathematical modeling

We consider a standard single machine-load system, as shown in Fig. 6, including the dynamics of the Automatic Voltage Regulator (AVR). E, E' and E_{fd} denote the voltage at the load bus, the voltage behind the generator's transient reactance, and the field excitation respectively. The voltage dynamic behavior for this network was studied in detail in [41], and it was assumed to be isolated from the frequency dynamics. The objective of the AVR control loop is to regulate E_{fd} at a specified reference value E_r, using as feedback the measured value of E_g. Following [41], the system is represented by a set of differential equations that govern the response of E', E_{fd} and an algebraic equation that couples E' with E. The differential equation that describes the evolution of E' corresponds to a one axis generator model, whereas the equation for E_{fd} is due to the first degree model that was used for the control dynamics. The algebraic equation that couples E' with E emanates from the power flow balance equations at every bus of the network. Solving it with respect to E', we get the following system (see [41]).

$$\dot{E} = \left(-\frac{X - X_d}{T_d X'} g_1(E) + \frac{X_d - X_d'}{T_d X'} \frac{E^2}{g_1(E)} + \frac{X_d - X_d'}{T_d} \frac{Q(E)}{g_1(E)} + \frac{E_{fd}}{T_d}\right) \frac{\partial g_1}{\partial E}^{-1},$$

$$\dot{E}_{fd} = (-E_{fd} + E_{fd}^0) u_1 + (-g_2(E) + E_r) u_2, \tag{15}$$

where

$$g_1(E) = E' = \frac{1}{E}\sqrt{X'^2(P^2 + Q(E)^2) + 2X'Q(E)E^2 + E^4},$$

$$g_2(E) = E_g = \frac{1}{E}\sqrt{X^2(P^2 + Q(E)^2) + 2XQ(E)E^2 + E^4},$$

$$Q(E) = Q_0 + HE + BE^2.$$

P, Q are the active and reactive parts of the load, where the latter was assumed to be voltage dependent. It consists of a constant power source Q_0, a current source HE and an impedance load BE^2, where $H, B \in \mathbb{R}$ are constant coefficients. Variables X, X_d, X_d' denote the transmission reactance, the generator's d-axis reactance and transient reactance respectively, and $X' = X + X_d'$. T_d is the open-circuit transient time constant. For the voltage dependent load we considered $H = B = 0.1$, whereas variables $u_1 \in [0.1, 1]$, $u_2 \in [-10, 10]$ are gains treated as control inputs (i.e. $u = \left[u_1\ u_2\right]^T$) so as to regulate E_{fd} to its reference value E_r. In this case no disturbance inputs are present, although one could consider load, parameter uncertainty, etc. Numerical values for the remaining parameters were retrieved from [41]. Note that (15) becomes singular along $\partial g_1/\partial E = 0$, but this occurs for an unacceptably low voltage value, outside the region of interest.

We consider a case where due to a fault, one of the two lines that connect the generator with the load is tripped at $t_f = 4$ s. The line closes then automatically after the fault is cleared at $t_c = 5$ s. The overall system can be described by the two-state automaton of Fig. 7. Define $x = \left[x_1\ x_2\ x_3\right]^T = \left[E\ E_{fd}\ z\right]^T \in \mathbb{R}^3$, where z is a "timer", and is appended to the system dynamics to capture the timed transitions between the two discrete modes. The vector fields $f_1, f_2 \in \mathbb{R}^3$ represent (15), augmented with $\dot{x}_3 = 1$, with the difference that the value of the reactance X is doubled once the line is tripped in

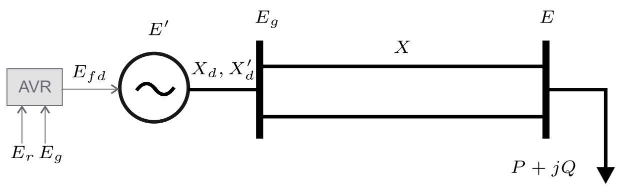

Fig. 6. Single machine-load power system with AVR.

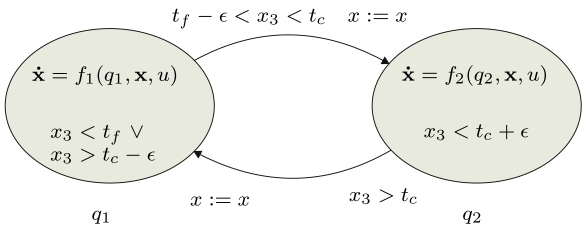

Fig. 7. Two-state hybrid automaton for the voltage control problem.

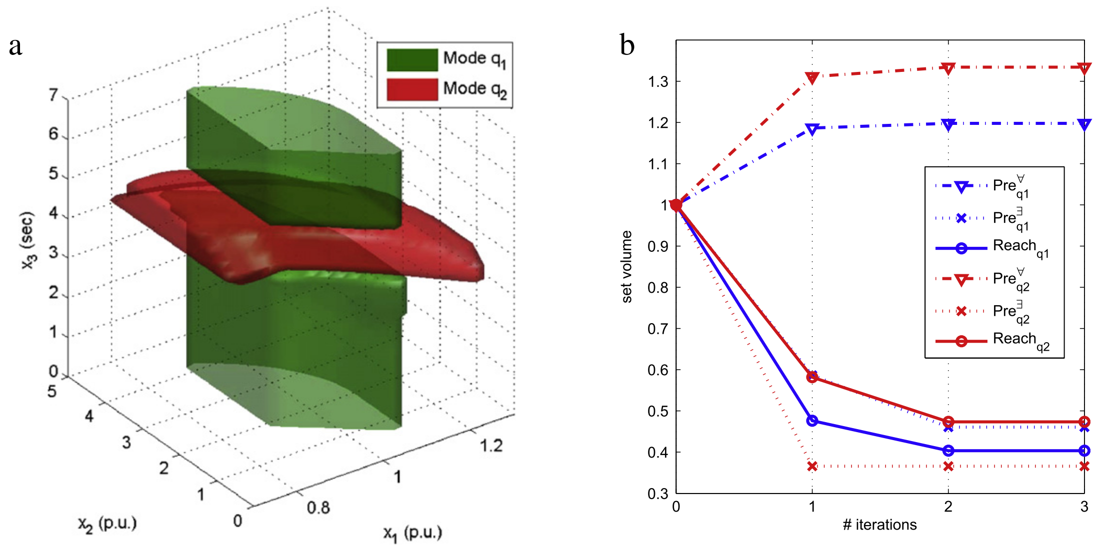

Fig. 8. (a) Hybrid discriminating kernel for each discrete mode. (b) Volume of $\text{Pre}^{\exists}(W_i)$, $\text{Pre}^{\forall}(W_i)$, and $\text{Reach}(\text{Pre}^{\exists}(W_i), \text{Pre}^{\forall}(W_i))$ for every iteration $i = 1, \ldots, 3$ and each mode. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

mode q_2. The hysteresis $\epsilon > 0$ is added to the guard and domain conditions of the automaton, to avoid Zeno phenomena and ensure that conditions 3 and 5 of Assumption 1 are satisfied.

4.2.2. Viability problem and simulation results

Considering the hybrid automaton of Fig. 7, the main objective is to determine the set of initial operating conditions, from which the system trajectories can start, and despite the line failure, there exists a control action such that the voltage remains within its safety limits both during the transient phase and after the reclosure of the line. A similar problem, but from a reachability perspective, was investigated in [26,27]. In these references the authors attempted to identify the time that the voltage exceeds the safety margins (i.e. voltage instability) after a fault. They represented the system by an acyclic graph since no line reclosure was considered, and hence a sequence of continuous calculations was applied instead. In our case the system is effectively also acyclic due to the timing constraints of the fault, and could be represented by a three-mode automaton whose third mode would be a sink state with the same continuous dynamics as the first one. Nevertheless, we keep the representation of Fig. 7 to illustrate some of the properties of the iterative procedure outlined in Algorithm 1.

The set $W_0 = \{q_1\} \times \{x \in \mathbb{R}^3 | 0.9 \leq x_1 \leq 1.1\} \cup \{q_2\} \times \{x \in \mathbb{R}^3 | 0.8 \leq x_1 \leq 1.2\}$ encodes the safety limits of E. Using the Level Set Method Toolbox [17] (version 1.1) on MATLAB 7.10 (at an Intel(R) Core(TM)2 Duo 2.66 GHz processor, 4 GB RAM, running Windows 7), we applied the viability algorithm, which reached a fixed point after two iterations, since at most two transitions may occur (i.e. $N = 2$). The overall procedure required 8.36 min. The continuous calculation at each mode was carried out until the viability sets had saturated (i.e. $T = \infty$). $\text{Viab}_{W_0}^{(N,T)}$ is then illustrated in Fig. 8(a). For any starting point in the low "green" region of q_1 there exists a control sequence, such that the corresponding trajectory remains safe until transiting to the safe part of q_2 ("red") at $x_3 = t_f$, and then return to the upper "green" region of q_1 at $x_3 = t_c$, remaining in W_0 for ever while evolving continuously.

Fig. 8(b) shows how the volume of $\text{Reach}(\text{Pre}^{\exists}(W_i), \text{Pre}^{\forall}(W_i))$, $\text{Pre}^{\exists}(W_i)$ and $\text{Pre}^{\forall}(W_i)$, computed as the number of grid points inside each set (normalized by the number of grid points inside the initial volume, using a $41 \times 41 \times 41$ grid), changes at every iteration $i = 1, \dots, 3$ for each mode. As expected (see Propositions 2 and 4 of [14] and Proposition 1), $\text{Reach}(\text{Pre}^{\exists}(W_i), \text{Pre}^{\forall}(W_i))$, $\text{Pre}^{\exists}(W_i)$ shrink, whereas the size of $\text{Pre}^{\forall}(W_i)$ increases with the number of iterations.

5. Concluding remarks

In this paper, we investigated the problem of computing viability sets for hybrid systems with competing inputs. Our analysis serves as the optimal control counterpart of [14], providing a complete characterization based entirely on optimal control and the definition of executions of hybrid automata, Different cases, based on whether the horizon of the continuous calculation and the number of discrete transitions was finite or infinite, were considered, and the algorithm was applied to a benchmark example and to the problem of voltage stability for a single machine-load system in case of a line fault.

Although the results reported here, together with those in the literature address a wide variety of viability type problems for hybrid systems, there are still a few open issues. The first is that no transitions forced by the inputs are allowed, hence a wide range of problems is excluded. Moreover, it was shown that the viability algorithm terminates at some ordinal number (see Section 3.2), but not necessarily the least one, which hampers the applicability of the method. Finally, for the infinite horizon case, the value function is semi-continuous, and hence uniqueness is no longer guaranteed. These issues constitute subject of current research.

Acknowledgments

We would like to thank the anonymous reviewers whose comments substantially improved the presentation of our results. In particular, we thank one of the reviewers for pointing out that the theoretical analysis of Section 3.2 is applicable to hybrid systems with executions continuing beyond the Zeno time.

Appendix

Proof of Proposition 2. Let $k \leq N$ denote the iteration at which the algorithm terminates. It suffices to show that $\text{Viab}_{W_0}^{(k,T)} = W_k$ for all $k \leq N$. If $k = N$ we directly have that $\text{Viab}_{W_0}^{(N,T)} = W_N$, whereas if $k < N$ notice that $\text{Viab}_{W_0}^{(N,T)} = W_N = W_k$, since we would have $W_i = W_k$ for all $i \geq k$. At the first part of the proof we show that $\text{Viab}_{W_0}^{(k,T)} \subseteq W_k$. To show that $W_k \subseteq \text{Viab}_{W_0}^{(k,T)}$ we use induction; the first step of the induction arguments is shown in Part 2 of the proof, whereas the induction proof is completed in Part 3. Finally in Part 4 we show that $\text{Viab}_F^{(N,T)} = \{(q, x) \in F | (q, x^z) \in \text{Viab}_{F\times[0,T]}^{(N,T)} \text{ and } z = 0\}$.

Part 1: We first show that $\text{Viab}_{W_0}^{(k,T)} \subseteq W_k$. Since $\text{Viab}_{W_0}^{(k,T)} \subseteq W_0$ and $W_k \subseteq W_0$, it suffices to show that $W_0 \setminus W_k \subseteq W_0 \setminus \text{Viab}_{W_0}^{(k,T)}$. Take $(\hat{q}, \hat{x}^z) \in W_0 \setminus W_k$. Fix any hybrid strategy (α, γ). Similarly to [14] we show that we can find an execution starting at $(q_0(\tau_0), x_0^z(\tau_0)) = (\hat{q}, \hat{x}^z)$ leaving W_0 after at most T units of continuous evolution and at most k discrete transitions. Since $(\hat{q}, \hat{x}^z) \notin W_k$, there exists $i < k$ such that $(\hat{q}, \hat{x}^z) \notin W_i = \text{Reach}(0, \text{Pre}^{\exists}(W_{i-1}), \text{Pre}^{\forall}(W_{i-1}))$. By Proposition 1 (part 2) we have that $(\hat{q}, \hat{x}^z) \notin \text{Pre}^{\exists}(W_{i-1})$. Therefore, either $\hat{x}^z \in \text{Dom}(\hat{q})$, or there exists $\hat{\delta}$ and $\hat{q}'$ such that $\hat{x}^z \in G(\hat{q}, \hat{q}')$ and $(\hat{q}', r^z(\hat{q}, \hat{q}', \hat{x}^z, \gamma(\hat{q}, \hat{x}^z), \hat{\delta})) \notin W_{i-1}$. In the latter case, set $\tau_0' = 0$, $q_1(\tau_1) = \hat{q}'$, $x_1^z(\tau_1) = r^z(\hat{q}, \hat{q}', \hat{x}^z, \gamma(\hat{q}, \hat{x}^z), \hat{\delta})$ and notice that $\tau_1 = 0$ and $(q_1(\tau_1), x_1^z(\tau_1)) \notin W_{i-1}$. If now $\hat{x}^z \in \text{Dom}(\hat{q})$, and since under Assumption 1 (part 3) $\text{Dom}(\hat{q})$ is open, there exists $d(\cdot)$ such that the solution $\phi(\cdot, \hat{q}, \hat{x}^z, \alpha, d)$ reaches $\text{Pre}^{\forall}(W_{i-1})$ without first reaching $\text{Pre}^{\exists}(W_{i-1})$. Since the admissible executions are limited to at most T time of continuous evolution, it suffices to consider the case where there exists $t_1 \in [0, T]$ such that $x^z(t_1) \in \text{Pre}^{\forall}(W_{i-1})$ and for all $t_2 \in [0, t_1)$, $x^z(t_2) \in \text{Dom}(\hat{q}) \setminus \text{Pre}^{\exists}(W_{i-1})$. Let $x_0^z(t) = x^z(t)$ for all $t \in [0, t_1]$. By the definition of $\text{Pre}^{\forall}$, either $(q_0(t_1), x_0^z(t_1)) \notin W_{i-1}$ or there exist $\hat{\delta}$ and $\hat{q}'$ such that $x_0^z(t_1) \in G(\hat{q}, \hat{q}')$ and $(\hat{q}', r^z(\hat{q}, \hat{q}', x_0^z(t_1), \gamma(\hat{q}, x_0^z(t_1)), \hat{\delta})) \notin W_{i-1}$. In the latter case, set $\tau_0' = t_1$, $q_1(\tau_1) = \hat{q}'$ and $x_1^z(\tau_1) = r^z(\hat{q}, \hat{q}', x_0^z(t_1), \gamma(\hat{q}, x_0^z(t_1)), \hat{\delta})$ and notice that $\tau_1 = t_1 \leq T < \infty$ and $(q_1(\tau_1), x_1^z(\tau_1)) \notin W_{i-1}$.

Overall, starting from $(\hat{q}, \hat{x}^z) \in W_0 \setminus W_i$ we constructed an admissible run that leaves W_{i-1} in less than T time of continuous evolution and after at most one discrete transition. Iterating i times we can construct a run that leaves W_0 in less than T time of continuous evolution and after at most $i \leq k$ discrete transitions. For every iteration the above arguments remain the same with the modification stated below. Assume that at iteration $j < k$ the system is at $(q_j(\tau_j), x_j^z(\tau_j)) = (\hat{q}, \hat{x}^z)$ with $z_j(\tau_j) = \tau_j > 0$. Everything remains the same apart from the case where $\hat{x}^z \in \text{Dom}(\hat{q})$. For an execution to be admissible we need to consider only the case where there exists $t_1 \in [0, T - \tau_j]$ so that the solution $\phi(\cdot, \hat{q}, \hat{x}^z, \alpha, d)$ reaches $\text{Pre}^{\forall}(W_{i-1})$ without first reaching $\text{Pre}^{\exists}(W_{i-1})$. Letting now $x_j^z(\tau_j + t) = x^z(t)$ for all $t \in [0, t_1]$, we can show as before that either $(q_j(\tau_j + t_1), x_j^z(\tau_j + t_1)) \notin W_{i-1}$ or $(q_{j+1}(\tau_{j+1}), x_{j+1}^z(\tau_{j+1})) \notin W_{i-1}$, for $\tau_{j+1} = \tau_j + t_1$. Hence for any hybrid strategy we have found discrete and continuous disturbance inputs such that the associated run starting from $(\hat{q}, \hat{x}^z)$, leaves W_0 via an admissible execution, which in turn implies that $(\hat{q}, \hat{x}^z) \notin \text{Viab}_{W_0}^{(k,T)}$.

Part 2: We will show that $W_1 \subseteq \text{Viab}_{W_0}^{(1,T)}$. We prove that if $(\hat{q}, \hat{x}^z) \in W_1$ then $(\hat{q}, \hat{x}^z) \in \text{Viab}_{W_0}^{(1,T)}$. Since $W_1 = \text{Reach}(0, \text{Pre}^{\exists}(W_0), \text{Pre}^{\forall}(W_0))$, for any $(q_0(\tau_0), x_0^z(\tau_0)) = (\hat{q}, \hat{x}^z) \in W_1$ we can distinguish two cases.

Case 1: For any $(\hat{q}, \hat{x}^z) \in W_1$ there exists an nonanticipative strategy α for the continuous controls such that for any continuous disturbance $d(\cdot) \in \mathcal{D}$, $(\hat{q}, x^z(t)) \in (\text{Pre}^{\forall}(W_0))^c \cap \text{Dom}(\hat{q})$ for all $t \in [0, T]$ (note that we are interested in executions with $\tau_0' - \tau_0 \leq T$). Choose then an arbitrary $\hat{t} \in [0, T]$. Therefore, there exist α such that for any d, $(\hat{q}, x^z(\hat{t})) \in W_0$ and there also exists $\upsilon \in V$ such that for all $\delta \in \Delta$ and $q' \in Q$ with $(\hat{q}, q') \in E$, $x^z(\hat{t}) \in G(\hat{q}, q')$ and $(q', r^z(\hat{q}, q', x^z(\hat{t}), \upsilon, \delta)) \in W_0$. Choose $\hat{q}' \in Q$, set $\tau_0' = \hat{t}$, $x_0^z(t) = x^z(t)$ for all $t \in [0, \hat{t}]$, $q_1(\tau_1) = \hat{q}'$, $\gamma(\hat{q}', x^z(t)) = \upsilon$ for all $t \in [0, \hat{t}]$, $x_1^z(\tau_1) = r^z(\hat{q}, \hat{q}', x^z(\hat{t}), \gamma(\hat{q}', x^z(\hat{t})), \delta)$, and notice that $\tau_1 = \hat{t}$ and $(q_1(\tau_1), x_1^z(\tau_1)) \in W_0$. Since $\hat{t} \in [0, T]$ was arbitrary, we have shown that there exists a hybrid strategy (α, γ), such that for any disturbance d and δ, all executions with $\tau_0' - \tau_0 \leq T$ starting from $(q_0(\tau_0), x_0^z(\tau_0)) \in W_1 \subseteq W_0$ are such that $(q_0(t), x_0^z(t)) \in W_0$ for all $t \in [\tau_0, \tau_0']$ and $(q_1(\tau_1), x_1^z(\tau_1)) \in W_0$. By Definition 4, the last statement implies that $(\hat{q}, \hat{x}^z) \in \text{Viab}_{W_0}^{(1,T)}$.

Case 2: For any $(\hat{q}, \hat{x}^z) \in W_1$ there exists an nonanticipative strategy α for the continuous controls such that for any continuous disturbance $d(\cdot) \in \mathcal{D}$, there exists $t_1 \in [0, T]$ such that $(\hat{q}, x^z(t_1)) \in \text{Pre}^{\exists}(W_0)$ and $(\hat{q}, x^z(t_2)) \in (\text{Pre}^{\forall}(W_0))^c \cap \text{Dom}(\hat{q})$ for all $t_2 \in [0, t_1]$. As in the previous case, we restricted $t_1 \in [0, T]$ since all admissible executions are such that $\tau_0' - \tau_0 \leq T$. Consider first any execution with $\tau_0' \in [0, t_1)$. Since $(\hat{q}, x^z(t)) \notin \text{Pre}^{\forall}(W_0)$ for all $t \in [0, \tau_0']$, following the same arguments as in Case 1 we can show that the system executions stay in W_0 via continuous evolution and one discrete transition, i.e. $(q_0(t), x_0^z(t)) \in W_0$ for all $t \in [\tau_0, \tau_0']$ and $(q_1(\tau_1), x_1^z(\tau_1)) \in W_0$. If $\tau_0' = t_1$, a transition is forced to occur since $(\hat{q}, x^z(\tau_0')) \in \text{Pre}^{\exists}(W_0)$. This implies that there exist a $\upsilon \in V$ such that for any $\delta \in \Delta$ and $q' \in Q$ with $(\hat{q}, q') \in E$, $x^z(\tau_0') \in G(\hat{q}, q')$ and $(q', r^z(\hat{q}, q', x^z(\tau_0'), \upsilon, \delta)) \in W_0$. Choose $\hat{q}' \in Q$, set $q_1(\tau_1) = \hat{q}'$, $x_0^z(t) = x^z(t)$ for all $t \in [0, t_1]$, $\gamma(\hat{q}', x^z(\tau_0')) = \upsilon$, $x_1^z(\tau_1) = r^z(\hat{q}, \hat{q}', x^z(\tau_0'), \gamma(\hat{q}', x^z(\tau_0')), \delta)$, and notice that $\tau_1 = t_1$ and $(q_1(\tau_1), x_1^z(\tau_1)) \in W_0$. The last statement implies that $(\hat{q}, \hat{x}^z) \in \text{Viab}_{W_0}^{(1,T)}$.

Part 3: We now show that $W_k \subseteq \text{Viab}_{W_0}^{(k,T)}$. To achieve this we will use induction. For $k = 1$ the claim follows from Part 2. Assume that the statement holds for some $j < k$, i.e. $W_j \subseteq \text{Viab}_{W_0}^{(j,T)}$. We should show that $W_{j+1} \subseteq \text{Viab}_{W_0}^{(j+1,T)}$. By the last part of Proposition 1,

$$\text{Reach}(0, \text{Pre}^{\exists}(W_j), \text{Pre}^{\forall}(W_j)) \subseteq \text{Reach}(0, \text{Pre}^{\exists}(\text{Viab}_{W_0}^{(j,T)}), \text{Pre}^{\forall}(\text{Viab}_{W_0}^{(j,T)})).$$

Since $W_{j+1} = \text{Reach}(0, \text{Pre}^{\exists}(W_j), \text{Pre}^{\forall}(W_j))$, it suffices to show that

$$\text{Reach}(0, \text{Pre}^{\exists}(\text{Viab}_{W_0}^{(j,T)}), \text{Pre}^{\forall}(\text{Viab}_{W_0}^{(j,T)})) \subseteq \text{Viab}_{W_0}^{(j+1,T)}.$$

Following the same arguments as in Part 2 with $\text{Viab}_{W_0}^{(j,T)}$ in place of W_0, we can show that there exist continuous controls such that for any disturbance input and any admissible execution starting from $(q_0(\tau_0), x_0^z(\tau_0)) \in \text{Reach}(0, \text{Pre}^{\exists}(\text{Viab}_{W_0}^{(j,T)}), \text{Pre}^{\forall}(\text{Viab}_{W_0}^{(j,T)})) \subseteq \text{Viab}_{W_0}^{(j,T)}$, $(q_0(t), x_0^z(t)) \in \text{Viab}_{W_0}^{(j,T)}$ for all $t \in [\tau_0, \tau_0']$ and $(q_1(\tau_1), x_1^z(\tau_1)) \in \text{Viab}_{W_0}^{(j,T)}$. Since $(q_1(\tau_1), x_1^z(\tau_1)) \in \text{Viab}_{W_0}^{(j,T)}$, following Definition 4 viability should be ensured for all executions with $n \leq j$ and $\sum_{i=1}^{n} \tau_i' - \tau_i \leq T$ (the first interval of those executions was assumed to be $[\tau_1, \tau_1']$). To achieve this, and since $W_0 = F \times [0, T]$, for all such executions the last component of the continuous state should not exceed T, i.e. $z_n(\tau_n) \leq T$ for all $n \leq j$. But $z_n(\tau_n) = \tau_n = \sum_{i=0}^{n} \tau_i' - \tau_i$. Therefore, all admissible executions that lead to $(q_1(\tau_1), x_1^z(\tau_1)) \in \text{Viab}_{W_0}^{(j,T)}$ should be restricted to one discrete transition and one interval $[\tau_0, \tau_0']$ of continuous evolution such that $\sum_{i=0}^{n} \tau_i' - \tau_i \leq T$ for all $n \leq j$.

Overall, starting from $(q_0(\tau_0), x_0^z(\tau_0)) \in \text{Viab}_{W_0}^{(j,T)} \subseteq W_0$ there exists a continuous control input such that for any disturbance, all executions with $n \leq j + 1$ (one transition is needed to reach $\text{Viab}_{W_0}^{(j,T)}$) and $\sum_{i=0}^{n} \tau_i' - \tau_i \leq T$ are such that $(q_i(t), x_i(t)) \in W_0$ for all $I_i \in \tau$ and all $t \in I_i$ with $i < n$, and $(q_n(\tau_n), x_n(\tau_n)) \in W_0$. By Definition 4 this implies that $(q_0(\tau_0), x_0^z(\tau_0)) \in \text{Viab}_{W_0}^{(j+1,T)}$ and concludes the induction proof.

Part 4: We will now show that $\text{Viab}_F^{(N,T)} = \{(q, x) \in F | (q, x^z) \in W_k \text{ and } z = 0\}$. Parts 1 and 3 lead to $W_k = \text{Viab}_{W_0}^{(k,T)} = \text{Viab}_{W_0}^{(N,T)}$, where $W_0 = F \times [0, T]$. Therefore, the set $W_k^t = \{(q, x) \in F | (q, x^z) \in W_k \text{ and } z = t\}$ for $t \in [0, T]$, denotes the states that remain in F following any execution restricted to $T - t$ time of continuous evolution. Notice that $\text{Viab}_F^{(N,T)}$ contains all states that remain in F for any execution restricted to T time of continuous evolution. Therefore $\text{Viab}_F^{(N,T)} = \bigcup_{t=0}^{T} W_k^t = W_k^0$. □

Proof of Lemma 1. *Part* 1: Let $x \in L$ such that $W_0 \geq x$ and $x \in \text{Post}(P)$, i.e. $P(x) \geq x$. Assume that for all $j \in \lambda$ with $j < i$, we have $W_j \geq x$. If i is a successor ordinal, then we have $W_{i-1} \geq x$. By Definition 6, $W_i = P(W_{i-1})$, and since $P(\cdot)$ is a monotone operator $P(W_{i-1}) \geq P(x)$. Therefore, $W_i = P(W_{i-1}) \geq P(x) \geq x$. If i is a limit ordinal, then by the induction hypothesis $\bigcap_{j<i} W_j \geq x$, and hence by Definition 6, $W_i \geq x$. Therefore, by transfinite induction, for all $i \in \lambda$, $W_i \geq x$.

Part 2: For all $i \in \lambda$ there exist unique a, b such that $i = a\omega + b$, with $a \geq i$ and $b < \omega$ [24]. If i is a limit ordinal, then $b = 0$ and for all $a' > a$, $i = a\omega < a'\omega$. But $a'\omega$ is a limit ordinal, hence by Definition 6, $W_i = \bigcap_{j<i} W_j \geq \bigcap_{j<a'\omega} W_j = W_{a'\omega}$. If $b \neq 0$, then i is a successor ordinal, and $i-1 = a\omega+b-1$. Assume now that for all $a' > a$ and for all $a'\omega \leq k \leq a'\omega+b-1$, we have $W_{i-1} \geq W_k$. Then, by monotonicity of P, we have $W_i = P(W_{i-1}) \geq P(W_k) = W_{k+1}$. Then, since $W_i \geq W_{a'\omega}$, with $k' = k+1$ we have that for all $a' > a$ and for all $a'\omega \leq k' \leq a'\omega + b$, $W_i \geq W_{k'}$. Therefore, by transfinite induction, the last statement concludes the proof. □

Proof of Lemma 2. Since $W_0 \in \text{Pre}(P)$, and P is a monotone operator on L, $\langle W_i, i \in \lambda\rangle$ is a decreasing chain. For the sake of contradiction, assume that this chain is strictly decreasing. By the definition of λ, this would imply that $\text{Card}(\langle W_i, i \in \lambda\rangle) = \text{Card}(\{i \in \lambda\}) > \text{Card}(L)$ ($\text{Card}(A)$ denotes the cardinality of A). On the other hand, we have that for all $i \in \lambda$, $W_i \in L$, hence $\text{Card}(\langle W_i, i \in \lambda\rangle) \leq \text{Card}(L)$. The last argument establishes a contradiction, and shows that $\langle W_i, i \in \lambda\rangle$ is a stationary decreasing chain. This implies that there exists $k \in \lambda$, such that $W_k = W_{k+1}$. Since $W_0 \geq W_k = W_{k+1} = P(W_k)$, W_k is a fixed point (and also a post-fixed point) of P less than or equal W_0. If k is a limit ordinal, denote $k = a\omega$. Then, for all $a' > a$ we have $a'\omega > a\omega$. By Definition 6, $W_{a'\omega} = \bigcap_{j<a'\omega} W_j = W_{a\omega} \cap \bigcap_{a\omega<j<a'\omega} W_j$. But $W_{a\omega} = W_k \in \text{Post}(P)$, so by Lemma 1 (part 2), for all $j \in \lambda$ (and also for all $j > a\omega$), $W_j \geq W_{a\omega}$. Therefore, $W_{a'\omega} = W_{a\omega}$ for all $a' > a$. Hence, the sequence $\langle W_{j\omega}, j \in \lambda\rangle$ is stationary decreasing (it is decreasing due to Lemma 1 (part 2)) with limit $W_{a\omega} = W_k$.

Let now $x \in L$ be such that $W_0 \geq x$ and $x \in \text{Post}(P)$. Then, by Lemma 1 (part 1), $W_k \geq x$, which implies that W_k is the greatest fixed point of P less than or equal W_0. Hence, $\text{gfp}(P) = \lim_P^l(W_0) = \vee\text{Post}(P)$. □

References

[1] H. Sussmann, Set-valued differentials and the hybrid maximum principle, in: IEEE Conference on Decision and Control, 2000, pp. 558–563.
[2] M. Shaikh, P. Caines, On the hybrid optimal control problem: analysis and algorithms, IEEE Transactions on Automatic Control 52 (9) (2007) 1587–1603.
[3] M. Boccadoro, Y. Wardi, M. Egerstedt, E. Verriest, Optimal control of switching surfaces in hybrid dynamical systems, Journal of Discrete Event Dynamic Systems 15 (4) (2005) 433–448.
[4] D. Liberzon, Calculus of Variations and Optimal Control Theory. A Concise Introduction, Princeton University Press, 2011.
[5] O. Maler, A. Pnueli, J. Sifakis, On the synthesis of discrete controllers for timed systems, in: Theoretical Aspects of Comp. Science, in: Ser. LNCS, vol. 900, 1995, pp. 229–242.
[6] M. Heymann, F. Lin, G. Meyer, Synthesis and viability of minimally interventive legal controllers for hybrid systems, Discrete Event Dynamic Systems: Theory and Applications 8 (2) (1998) 105–135.
[7] H.W. Toi, The synthesis of controllers for linear hybrid automata, in: Proc. IEEE Conf. Decision Control, San Diego, CA, 1997, pp. 4607–4613.
[8] J. Lygeros, C. Tomlin, S. Sastry, Controllers for reachability specifications for hybrid systems, Automatica 35 (1999) 349–370.
[9] C. Tomlin, J. Lygeros, S. Sastry, A game theoretic approach to controller design for hybrid systems, Proceedings of the IEEE 88 (7) (2000) 949–969.
[10] J. Aubin, Viability Theory, Birkhäuser, Boston, 1991.
[11] T. Basar, G.J. Olsder, Dynamic Noncooperative Game Theory, Academic Press, New York, 1982.
[12] J. Lygeros, On reachability and minimum cost optimal control, Automatica 40 (6) (2004) 917–927.
[13] I. Mitchell, A.M. Bayen, C. Tomlin, A time-dependent Hamilton–Jacobi formulation of reachable sets for continuous dynamic games, IEEE Transactions on Automatic Control 50 (2005) 947–957.
[14] Y.G. Gao, J. Lygeros, M. Quincampoix, On the reachability problem for uncertain hybrid systems, IEEE Transactions on Automatic Control 52 (9) (2007) 1572–1586.
[15] J.P. Aubin, J. Lygeros, M. Quincampoix, S. Sastry, N. Seube, Impulse differential inclusions: a viability approach to hybrid systems, IEEE Transactions on Automatic Control 47 (1) (2002) 2–20.
[16] K. Margellos, J. Lygeros, Revisiting the viability algorithm for hybrid systems using optimal control, in: Proceedings of IFAC Conference on Analysis and Design of Hybrid Systems, 2012, pp. 25–32.
[17] I.M. Mitchell, Application of level set methods to control and reachability problems in continuous and hybrid systems, Ph.D. Dissertation, Stanford University, 2002.
[18] J. Ding, C. Tomlin, Robust reach-avoid controller synthesis for switched nonlinear systems, in: IEEE Conference on Decision and Control, 2010, pp. 6481–6486.
[19] K. Margellos, J. Lygeros, Hamilton–Jacobi formulation for reach-avoid differential games, IEEE Transactions on Automatic Control 56 (8) (2011) 1849–1861.
[20] I.J. Fialho, T. Georgiou, Worst case analysis of nonlinear systems, IEEE Transactions on Automatic Control 44 (6) (1999) 1180–1196.
[21] A.B. Kurzhanski, I.M. Mitchell, P. Varaiya, Optimization techniques for state-constrained control and obstacle problems, Journal of Optimization Theory and Applications 128 (3) (2006) 499–521.
[22] A. Tarski, A Lattice-theoretical fixpoint theorem and its applications, Pacific Journal of Mathematics 5 (1955) 285–309.
[23] B. Davey, H. Priestley, Introduction to Lattices and Order, second ed., Cambridge University Press, 2002.
[24] P. Cousot, R. Cousot, Constructive versions of Tarski's fixed point theorems, Pacific Journal of Mathematics 82 (1) (1979) 43–57.
[25] A. Nerode, J.B. Remmel, A. Yakhnis, Controllers as fixed points of set-valued operators, in: Hybrid Systems: Computation and Control, Springer-Verlag, 1995, pp. 344–358.
[26] Y. Suzuki, T. Hikihara, Predicting voltage instability of power system via hybrid system reachability analysis, in: American Control Conference, 2007, pp. 4166–4171.
[27] E. Cross, I. Mitchell, Level set methods for computing reachable sets of systems with differential algebraic equation dynamics, in: American Control Conference, 2008, pp. 2260–2265.
[28] J. Lygeros, K. Johansson, S. Simic, J. Zhang, S. Sastry, Dynamical properties of hybrid automata, IEEE Transactions on Automatic Control 48 (1) (2003) 2–17.
[29] L. Evans, P. Souganidis, Differential games and representation formulas for solutions of Hamilton–Jacobi–Isaacs equations, Indiana University of Mathematics Journal 33 (5) (1984) 773–797.
[30] P.P. Varaiya, On the existence of solutions to a differential game, SIAM Journal on Control and Optimization 5 (1) (1967) 153–162.
[31] M. Bardi, I. Capuzzo-Dolcetta, Optimal Control and Viscosity Solutions of Hamilton–Jacobi–Bellman Equations, Birkhäuser, Boston, 2008.
[32] P. Cardaliaguet, Nonsmooth semipermeable barriers, Isaacs equation, and application to a differential game with one target and two players, Applied Mathematics and Optimization 36 (2) (1997) 125–146.
[33] L. Evans, Partial Differential Equations, American Mathematical Society, Providence, RI, 1998.
[34] J.A. Sethian, Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Materials Science, Cambridge University Press, New York, 1996.

[35] I. Mitchell, Demonstrating numerical convergence to the analytic solution of some backwards reachable sets with sharp features, UBC Department of Computer Science, Technical Report TR-2004-01.
[36] I. Mitchell, Comparing forward and backward reachability as tools for safety analysis, in: Hybrid Systems: Computation and Control, 2007, pp. 428–443.
[37] H. Zheng, E. Lee, A. Ames, Beyond Zeno: Get on with it! in: Proceedings of Hybrid Systems: Computation and Control, HSCC, in: LNCS, vol. 3927, 2006, pp. 1–15.
[38] K. Johansson, M. Egerstedt, J. Lygeros, S. Sastry, Regularization of Zeno hybrid automata, Systems and Control Letters (1999) 141–150.
[39] G. Birkhoff, Lattice Theory, third ed., in: Colloquium Publications, vol. 25, Amer. Math. Soc., Providence, Rhode Island, 1976.
[40] E. Barron, R. Jensen, Semicontinuous viscosity solutions for Hamilton Jacobi equations with convex Hamiltonians, Communications in Partial Differential Equations 15 (12) (1990) 1713–1742.
[41] V. Venkatasubramanian, H. Schättler, J. Zaborszky, Voltage dynamics: study of a generator with voltage control, transmission, and matched MW load, IEEE Transactions on Automatic Control 37 (11) (1992) 1717–1733.