

---

## Literature Review

### *Hybrid Systems: Who, What, When?*

In this chapter, we review existing literature on hybrid systems that is related to this work. This is carried out by first considering three specific approaches to viability of hybrid systems, these being due to Nerode and colleagues, Aubin and colleagues and Deshpande–Varaiya. This review is carried out in Sections 2.1–Section 2.3. In Section 2.4, literature related specifically to Chapters 3–8 is then reviewed. Some concluding remarks are made in Section 2.5. Given that hybrid systems is a field that encompasses a variety of problem domains and disciplines, there exists a vast body of related work. Although we do provide some general overview of the field, we will focus on approaches to hybrid systems, and in particular their control, that have the most direct impact on this work.

### 2.1 Nerode *et al* Approach to Viability of Hybrid Systems [49], [70]

In [49], the fundamental problem which is tackled is to extract, given a continuous plant simulation model and performance specification on plant state trajectories, a finite automaton which forces the hybrid system to satisfy the performance specification. More specifically, they wish to capture conditions under which finite state control automata exist which ensure viability is satisfied. The situation is such that Kohn *et al* assume that they are given

a continuous feedback control function which enforces viable trajectories for a plant and that the objective is to investigate how to extract finite automaton which exhibit controllable and observable behaviour and enforce the same viability as in the continuous time case.

In this work, all hybrid systems are assumed to be simple hybrid systems with fixed control intervals. A simple hybrid system runs open loop within a time interval  $[n\Delta, (n+1)\Delta]$  based on a control function  $c_n$  and disturbance  $d_n$  supplied at time  $n\Delta$ . The control automaton receives as input at time  $n\Delta$  the current state  $x$  of the plant, then runs open loop with no further inputs until the time  $(n+1)\Delta$ . Based on its state at time  $(n+1)\Delta$ , the control automaton transmits a new control function  $c_{n+1}$  to the plant to be used for the time interval  $[(n+1)\Delta, (n+2)\Delta]$  and this process repeats. It is assumed that the plant is described by a vector first order differential equation

$$\dot{x} = f(x, c, d)$$

with parameters  $c, d$ . It is further assumed that

$$\dot{x}(t) = f(x, \tilde{c}(t), \tilde{d}(t))$$

is such that for any time  $t_0$ , for any initial state  $x(t_0)$ , for any admissible control and disturbance functions  $\tilde{c}(t), \tilde{d}(t)$  defined on  $[0, \infty]$ , there is a unique solution  $x(t)$  defined on  $[0, \infty]$  satisfying the differential equation.

The initial value of  $\dot{x}(t)$  for the interval  $[n\Delta, (n+1)\Delta]$  is not inherited from the previous interval, but is computed from the differential equation based on the current plant state, the initial value of the new control function and the new disturbance at  $n\Delta$ . This results in the vector field changing direction abruptly at time  $n\Delta$  which is characteristic of hybrid control.

Since the plant differential equation is taken as autonomous in this work, the behaviour in an interval of length  $\Delta$  is the translate of the behaviour in any other interval of length  $\Delta$ . Therefore,  $c(t)$  and  $d(t)$  will be assumed to be defined on  $[0, \Delta]$  and translate them by  $n\Delta$  for use on the interval  $[n\Delta, (n+1)\Delta]$ .

**Definition 2.1.** *The continuous plant induces an automaton, which is called the  $\Delta$ -plant automaton associated with a simple hybrid system. It has two input alphabets, the set  $D$  of admissible disturbance functions and the set  $C$  of admissible control functions. Its set of internal states is the set of plant states. Its transition function assigns to input letters control  $c(t)$  and disturbance  $d(t)$  and automaton current state  $s_0$ , the new automaton state  $x(\Delta)$  where  $x(t)$  is a plant state trajectory such that  $x(0) = s_0$  is the solution to the differential equation*

$$\dot{x} = f(x(t), c(t), d(t)).$$

The viability set is denoted as  $VS$ , a subset of plant states which is usually assumed to be closed and compact.

**Definition 2.2.** *A trajectory  $x(t)$  over an interval of time  $[0, \Delta]$  is called viable if for all  $t$  in that interval,  $x(t) \in VS$ . Similarly, a trajectory extending over  $[0, \infty]$  is viable if for all  $n \geq 0$ , the trajectory  $x_n(t) = x(t - n\Delta)$  over  $[0, \Delta]$  is viable.*

Associated with viability of a simple hybrid system are three definitions of local graphs given next.

**Definition 2.3.** *The abstract viability graph is an obvious analogue to the viability kernels of continuous time systems. Non-empty closed compact subsets of this graph and closed viability sets lead to finite automata that enforce viability.*

**Definition 2.4.** *The sturdy local viability graph is such that non-empty closed compact subsets lead to finite automata that force viability and also are “safe” under small errors in state and control measurements.*

**Definition 2.5.** *The e-sturdy local viability graph represents those hybrid systems with a sensor of plant states with error bounded by a fixed  $e$ . This leads to finite state control automata whose analog to digital converter, or sensor of plant state has error bounded by  $e$  and also enforces viability.*

The latter two graphs are not developed extensively in [49] and so will not be considered here in any detail. Next, definitions for nodes and edges of the abstract local viability graph are given.

**Definition 2.6.** *The nodes of the local viability graph are those pairs*

$$(c_0, s_0) \in C \times VS$$

*such that for any disturbance  $d_0 \in D$ , the trajectory  $x(t)$  determined by  $d_0$ , control  $c_0$  and initial state  $s_0$  is viable.*

**Definition 2.7.** *There is a directed edge from node  $(c_0, s_0)$  to node  $(c_1, s_1)$  if and only if*

1.  $(c_0, s_0)$  and  $(c_1, s_1)$  both nodes of the local viability graph and
2. *There is a disturbance  $d_0 \in D$  such that the trajectory  $x(t)$  with disturbance  $d_0$  and control  $c_0$  and initial condition  $x(0) = s_0$  has  $x(\Delta) = s_1$ .*

The pair  $(c_0, s_0)$  is referred to as the *tail* and  $(c_1, s_1)$  is referred to as the *head* of the directed edge. There may be nodes of the abstract local viability graph that are not heads nor tails of any edges. In this case, these nodes are dropped at the beginning of the construction of the graph.

For the local viability automaton

- The input alphabet is the set of viable plant states  $VS$ .
- The states are the set of controls.
- The non-deterministic transition relation maps a pair  $(c_0, s_1) \in C \times VS$  to a control  $c_1$  if and only if there exists an edge in the abstract local viability graph with tail  $(c_0, s_0)$  and head  $(c_1, s_1)$ . This is a partially defined transition relation. The interpretation is that  $c_0$  should be thought of as the control used in the previous control interval which has, due to a disturbance, produced the current plant state  $s_1$ . Then, with input letter  $s_1$  when in local viability automaton state  $c_0$ , the local viability automaton moves to state  $c_1$  and outputs letter  $c_1$ .

Next we consider viability over the interval  $[0, \infty]$ . Assume we are given a directed graph  $T$  which consists of a non-empty set  $T$  of nodes and a subset  $E$  of  $T \times T$  of its directed edges such that

each node is incident on at least one edge. Each subset  $T'$  of  $T$  defines a subgraph with edges  $E' = E \cap (T' \times T')$ . A path is a finite or infinite sequence of edges such that the head of each edge is the tail of the next edge. An end node of a graph is a node which is not the tail of any edge in that graph. Let  $P(T)$  denote the power set of  $T$ .

**Definition 2.8.** *Suppose graph  $T$  is given.*

1. *Define a monotone decreasing operator  $F : P(T) \rightarrow P(T)$  by letting  $F(T')$  be the set of nodes of  $T'$  which are not end nodes of  $T'$  and which are on at least one edge of  $T'$ .*
2. *For each ordinal  $\alpha$ , define an operator  $F^\alpha : P(T) \rightarrow P(T)$  by transfinite induction as*
  - a)  $F^0(T') = F(T')$ ,
  - b)  $F^{\alpha+1}(T') = F(F^\alpha(T'))$ ,
  - c)  $F^\lambda(T') = \bigcap_{\alpha > \lambda} F^\alpha(T')$  if  $\lambda$  is a limit ordinal.

**Proposition 2.9.** *Suppose that  $T' \subseteq T$ .*

1. *Then  $T'$  is a fixed point of  $F$  if and only if every node of  $T'$  is the initial node of some infinite path in  $T'$ .*
2. *There is a least ordinal  $\alpha$  such that*

$$F^{\alpha+1}(T') = F^\alpha(T')$$

3. *If  $\alpha$  is the least ordinal such that  $F^{\alpha+1}(T') = F^\alpha(T')$ , then  $F^\alpha(T')$  is the largest fixed point of  $F$  contained in  $T'$ .*

**Proposition 2.10.** *Suppose that:*

1. *The nodes of  $T$  are elements of a separable metric space and*
2.  *$T'$  is a subgraph of  $T$  such that for every ordinal  $\alpha$  and for every end node of  $F^\alpha(T')$  or node on no edge of  $F^\alpha(T')$ , there is a neighbourhood containing that node and no other node of  $F^\alpha(T')$ . (Here we interpret  $F^0$  to be the identity map on  $P(T)$ .)*

*Then*

1. *The least ordinal  $\alpha$  such that  $F^\alpha(T') = F^{\alpha+1}(T')$  is a countable ordinal.*
2. *If  $T'$  is closed, then  $F(T')$  is closed.*

3. If  $T'$  is closed, then so are all  $F^\alpha(T')$ .
4. If  $T'$  is closed, then so is the maximal fixed point of  $F$  under  $T'$ .

Next the abstract viability graph of a simple hybrid system with control intervals  $[n\Delta, (n+1)\Delta]$ ,  $n = 0, 1, \dots$  with viability set  $VS \subset S$  is defined.

**Definition 2.11.** *The abstract viability graph is the kernel of the abstract local viability graph.*

**Definition 2.12.** *The abstract viability automaton has a transition corresponding to each edge of the abstract viability graph with tail  $(c_0, s_0)$  and head  $(c_1, s_1)$ .*

**Definition 2.13.** *A control policy can be defined as a map on  $C \times S$  to  $P(C)$  which assigns to a pair  $(c_0, s_1)$  the set of choices of  $c_1$ , any of which is permitted under the control policy. Alternately, a control policy is simply a subset  $CP$  of  $C \times S \times C$  consisting of triples  $(c_0, s_1, c_1)$ .*

The largest control policy is the universal policy  $C \times S \times C$  which permits any choice of  $c_1$ . The smallest control policy is the null policy which is devoid of choice.

**Definition 2.14.** *An edge of the abstract local viability graph with tail  $(c_0, s_0)$  and head  $(c_1, s_1)$  is an abstract policy edge for policy  $CP$  if  $(c_0, s_1, c_1)$  is in  $CP$ . The policy graph for a policy consists of its policy edges. A path consisting of abstract policy edges is an abstract policy path for  $CP$ . (An abstract policy path is just a path in the abstract local viability graph that can arise by following the policy.)*

**Proposition 2.15.** *The trajectories on  $[0, \infty]$  produced by a control policy and admissible disturbances on  $[0, \infty]$  are all viable if and only if all infinite policy paths are abstract policy paths.*

Next, results on the closure of the abstract local viability graph are given.

**Corollary 2.16.** *Suppose we are given a simple hybrid system, a  $\Delta$ , and a closed viability set  $VS$ . Suppose that  $S, C$  are separable metric spaces, and that the set  $T$  of nodes of the abstract local viability graph is closed. Then the set of nodes of the abstract viability graph is also closed.*

**Proposition 2.17.** *Suppose that  $VS$  is closed. Suppose that for any fixed  $t_0$  with  $0 \leq t_0 \leq \Delta$  and any disturbance  $d$  in  $D$ , the map  $(s_0, c_0) \rightarrow x(t_0)$  with domain  $S \times C$  is continuous. Then the set of nodes of the abstract local viability graph is closed.*

**Proposition 2.18.** *Suppose we are given a simple hybrid system, a  $\Delta$ , and a closed viability set  $VS$ . Suppose that  $S, C$  are separable metric spaces, and that the set  $T$  of nodes of the abstract local viability graph is closed. Moreover, suppose that every closed subgraph  $T'$  of  $T$  has that property that for every end node of  $T'$  or node on no edge of  $T'$ , there is a neighbourhood containing that node and no other node of  $T'$ . Then the set of nodes of the abstract local viability graph is also closed.*

**Proposition 2.19.** *Suppose that*

1.  *$VS$  is closed and for any fixed  $t_0$  with  $0 \leq t_0 \leq \Delta$  and any disturbance  $d$  in  $D$ , the map  $(s_0, c_0) \rightarrow x(t_0)$  with domain  $S \times C$  is continuous.*
2.  *$C$  and  $D$  are compact and the map  $(s_0, c_0, d_0) \rightarrow x(\Delta)$  with domain  $S \times C \times S$  is continuous.*

*Then the set of nodes of the abstract local viability graph  $T$  is closed and every closed subgraph  $T'$  of  $T$  has the property that for every end node of  $T'$  or node on no edge of  $T'$ , there is a neighbourhood containing that node and no other node of  $T'$ .*

The abstract viability graph, if non-empty is a non-deterministic automaton which enforces viability if started on a node of the abstract viability graph. Viability could be ensured for  $[0, \Delta]$  if we knew how to implement this automaton as long as the automaton is started in state  $s_0$  with control  $c_0$  such that there is node  $(c_0, s_0)$  of the abstract viability graph. The problem is that this automaton has been arrived at in a non-constructive manner generally having a highly non-constructive transition relation.

Lastly, the next proposition provides conditions for the existence of a finite state control automaton that ensures viable trajectories over  $[0, \Delta]$  when the set of controls is finite.

**Proposition 2.20.** *Suppose that  $S$  is the set of plant states,  $C$  is the set of controls, and  $VS$  is the set of viable states. Suppose also that*

1.  *$R$  is a non-empty closed subset of the abstract viability graph.*
2. *For any  $(\bar{c}_0, \bar{s}_0) \in R$  and any disturbance  $d \in D$ , if  $x(t)$  is the resulting trajectory, there exists a  $\bar{c}_1 \in C$  such that  $(\bar{c}_1, x(\Delta)) \in R$ . (Note that this says that  $R$  is a fixed point of the operator  $F$ .)*
3. *The spaces  $S, C$  are compact metric spaces.*
4. *The viability set  $VS$  is a closed subset of  $S$ .*
5. *Let  $R_0, R_1$  be, respectively, the projections of  $R$  on its first coordinate  $c_0$  and on its second coordinate  $s_0$ . Assume that  $R$  has the following “sturdiness property”. For any  $r = (c_0, s_0)$  in  $R$ , there exists a pair of open sets  $U_r \subseteq C$ ,  $V_r \subseteq VS$ , such that  $(c_0, s_0) \in U_r \times V_r$  and  $(U_R \cap R_0) \times (V_r \cap R_1) \subseteq R$ .*

*Then there exist finite state automata which, regarded as control policies, have infinite policy paths which are policy paths of  $R$ . That is, there are finite state automata which can produce viable trajectories from certain initial conditions no matter what the disturbance. (We do not assert that every policy path of  $R$  is a policy path for the automaton.)*

In [70], the problem of constructing a controller for a hybrid system which solves the viability problem is considered. This work is a continuation of the Nerode–Kohn program whereby they take a top-down approach to design of hybrid controllers. This top-down approach investigates methods of finding finite control automaton for mode switching from performance specifications. It is argued that part of every performance specification on the various constraints on the plant trajectory. The constraint of interest in this work is taken as a viability constraint whereby the constraint restricts the plant trajectory to stay forever in a specified set  $VS$  of plant states. The set  $VS$  is often called the *viability set*.



The continuous plant dynamics are given by an ordinary vector differential equation

$$\dot{x} = f(x, c, d)$$

with  $x, c, d$  being finite dimensional vectors belonging to some Euclidean spaces. It is assumed that there is a unique  $x(t)$  defined on the interval  $[t_0, t_0 + \Delta]$  with this function called the plant state trajectory which starts at  $x(t_0)$  and which is determined by  $c(t)$  and  $d(t)$ .

The plant dynamics can be described by a differential inclusion by eliminating the dependence of the plant state on disturbances by letting  $D$  be the admissible set of values of the disturbance  $d$  and letting  $F(x, c) = \{f(x, c, d) : d \in D\}$ . By a lemma of Fillipov, an equivalent plant description is given by

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

The inclusion preserves information and also suggests that we do not know exactly how the disturbances effect  $x(t)$  which is the case in practice. Therefore, it will be assumed that the plant is modelled as a differential inclusion. Since plants modelled by differential equations are common, some of the results will also be given based on this plant formalism.

A solution to a differential inclusion is an absolutely continuous function  $x(t)$  satisfying the inclusion almost everywhere over a time interval. Assumptions on the righthand side of the differential inclusion are used to insure the following:

- (I) Existence of a plant trajectory for any point of  $VS$  and for any admissible control law inserted into the differential inclusion.
- (II) Continuous dependence of plant trajectories on initial plant state. That is, suppose we are given a fixed control law  $c$  and a sequence  $x_n$  of plant states converging to  $x$ . Then, for any plant trajectory  $y(t)$  which begins at  $x$  and is guided by  $c$ , there exists a sequence of plant trajectories  $y_n(t)$  which begin at  $x_n$  and are guided by  $c$  and which converges to  $y(t)$  uniformly over the sampling interval. (All plant trajectories take their values from the viability set  $VS$ .)

Assumptions on  $F$  ensuring (I) are for example:

1.  $F$  is upper or lower semicontinuous, and
2. the values of  $F$  are compact, convex sets.

For the remainder of the discussion, premises of the theorems will be stated in terms of properties (I) and (II).

Control laws need only to be defined over the time interval  $[0, \Delta]$  so that control strategies need output a control law only over the interval  $[0, \Delta]$ .

Suppose that we are given a subset  $VS$  of plant state space. The viability problem is to find a subset  $VS'$  of  $VS$  and a control strategy which will ensure that if a plant state is initially in  $VS'$ , then the plant trajectories guided by that strategy will stay in  $VS$  forever.

**Definition 2.21.** *Suppose that we are given a plant with viability set  $VS$ , and a set of admissible control laws  $W$  over the time interval  $[0, \Delta]$ . A control automaton for such a plant is a nondeterministic input-output (Mealy) automaton  $(V, W, Q, TO, q_{in}, PS')$  consisting of the following:*

- (CA1) *Its input alphabet is the set of measurements of the plant states  $V$ .*
- (CA2) *Its output alphabet is a subset of  $W$ .*
- (CA3) *Its set of states is a discrete set  $Q$ .*
- (CA4) *Its transition-output function is a set-valued function  $TO$  whose graph is a subset of  $Q \times V \times Q \times W$ .*
- (CA5) *Its initial internal state is  $q_{in} \in Q$ .*
- (CA6) *A set  $PS'$  is the set of admissible initial states of the plant.*

**Definition 2.22.** *An infinite plant trajectory guided by a control automaton is a concatenation of a sequence of plant trajectories  $x_n(t)$  over intervals  $[n\Delta, (n+1)\Delta]$  with the following properties. (These are referred to as “parts” of the infinite trajectory.)*

1. *The initial trajectory  $x_0$  begins at some admissible plant initial state  $x_{in}$  from the  $VS'$  and it either satisfies the differential inclusion  $\dot{x}_0 \in F(x_0, u_0)$  if the plant is modeled by the differential inclusion  $\dot{x} \in F(x, u)$ , or there is an admissible disturbance  $d_0$  such that it is the trajectory determined by*

$\dot{x}_0 = f(x_0, u_0, d_0)$  if the plant is modeled by a vector differential equation  $\dot{x} = f(x, u, d)$  where  $u_0$  is an initial control law such that  $(q_{in}, v_{in}, q', u_0)$  is in the graph of  $TO$  for some measurement  $v_{in}$  of  $x_{in}$  and state  $q'$ .

2. The part of the trajectory numbered  $n + 1$  begins at the end of the trajectory part numbered  $n$ .
3. The  $(n + 1)$ -th part of the trajectory  $x_{n+1}$  either satisfies the differential inclusion  $\dot{x}_{n+1} \in F(x_{n+1}, u_{n+1})$  if the plant is modeled by the differential inclusion  $\dot{x} \in F(x, u)$  or there is an admissible disturbance  $d_{n+1}$  such that it is the trajectory determined by  $\dot{x}_{n+1} = f(x_{n+1}, u_{n+1}, d_{n+1})$  if the plant is modeled by a vector differential equation  $\dot{x} = f(x, u, d)$  where  $u_{n+1}$  is the control law output by the control automaton for a measurement of the plant state at time  $(n + 1)\Delta$  and control automaton state  $q_n$  of the endpoint of the  $n$ -th part of the trajectory.

A control automaton is *correct* with respect to  $VS$  and the plant model if every plant trajectory guided by the automaton lies in the set  $VS$ .

Next a controllability operator over subsets of  $VS$  is defined. We say that a measurement  $m$  corresponds to a subset  $Z$  of  $VS$  if  $m \in B(Z, e)$  where  $e$  is the measurement error.

**Definition 2.23.** Consider a set  $Z \subset VS$ . Call a subset  $Z'$  of  $Z$  controllable with target  $Z$  if for any measurement  $v$  corresponding to the subset  $Z'$ , there exists a nonempty subset  $W'$  of the set of admissible control laws  $W$  such that for any point  $x$  of  $Z'$  whose measurement is  $v$ , there is a control law  $w \in W'$  ensuring the following

- (CTI1) Any plant trajectory which starts at  $x$  and is guided by the control law  $w$  will end in the set  $Z$  at the end of the sampling interval.
- (CTI2) Any plant trajectory which starts at  $x$  and is guided by the control law  $w$  remains in the set  $VS$  throughout the interval.

**Definition 2.24.** The value of the controllability operator on a set  $Z \subseteq VS$  is the largest subset of  $Z$  which is controllable with target  $Z$ .

The controllability operator will be denoted by  $H$ . The above definition requires  $H$  to be monotone decreasing.

Next, we state the main result on existence of a correct finite state control automaton. For this result, we only need the existence of plant trajectories beginning at any point of  $VS$  for any admissible control law.

**Theorem 2.25.** *Suppose that*

- 1 *the plant is given by a differential inclusion has only a finite set of control laws  $C$  and satisfies condition (I) above and that*
- 2 *the plant state measurements consist of all values which deviate from the actual plant states in the viability set  $VS$  by less than  $\epsilon > 0$ .*

*Assume that a viability set  $VS$  and the length of the sampling interval  $\Delta$  are given. Then there exists a correct finite state control automaton if and only if all of the following conditions hold.*

- (A) *The corresponding controllability operator has a non-empty fixed point  $V$ .*
- (B) *There exists a sequence of subsets of  $V$ ,  $(V(1), \dots, V(n))$  and a sequence of subsets of  $C$ ,  $(C(1), \dots, C(n))$  such that  $V = \bigcup_{i=1}^n V(i)$  and for any  $k$  with  $1 \leq k \leq n$ , if  $m$  is a measurement for  $V(k)$ , then there is a control law  $c \in C(k)$  corresponding to  $m$  and an index  $j$  such that any plant trajectory which starts at an  $x \in V(k)$  whose measurement is  $m$  and which is guided by  $c$  ends in  $V(j)$  at the end of the sampling interval and stays entirely in  $VS$ .*

For the next theorem we assume that the plant is modelled by a vector differential equation  $\dot{x} = f(x, c, d)$  which has the following properties analogous to (I) and (II) above for the differential inclusion case.

- (I\*) *For every point in the set  $VS$  and any pair consisting of an admissible control law and an admissible disturbance function, there is a unique plant trajectory beginning from that point and guided by the control law and the disturbance function which exists over entire sampling interval and*

(II\*) The above plant trajectories depend continuously on their starting point and are contained in  $VS$ .

**Theorem 2.26.** *Suppose that*

1. *the plant is modelled by a vector differential equation  $\dot{x} = f(x, c, d)$  satisfying conditions (I\*) and (II\*) and*
2. *the plant state measurements consist of all values which deviate from the actual plant states in the viability set  $VS$  by less than  $\epsilon > 0$ .*

*If the viability set  $VS$  is closed, then the closure of any fixed point of the controllability operator is itself a fixed point of that same operator.*

**Theorem 2.27.** *Suppose that*

1. *the plant is modelled by a vector differential equation  $\dot{x} = f(x, c, d)$  satisfying conditions (I\*) and (II\*) and*
2. *the plant state measurements consist of all values which deviate from the actual plant states by less than  $\epsilon > 0$ .*

*If a set  $Z$  is closed, then the closure of any subset of  $Z$  which is controllable with target  $Z$  is also a subset of  $Z$  which is controllable with target  $Z$ .*

The transition–output function of a control automaton is often given in practice by inequalities separating measurements at which transitions occur. It is desirable if the correctness of the automaton does not depend on whether a strict inequality is replaced by a non–strict one since in practice exact values cannot be measured.

**Theorem 2.28.** *Suppose that the plant is either modelled by a differential inclusion  $\dot{x} \in F(x, c)$  satisfying condition (I) or is modelled by a vector differential equation  $\dot{x} = f(x, c, d)$  satisfying condition (I\*). Moreover assume that the set of plant state measurements  $M$  consist of all values which deviate from the actual plant states in the viability set  $VS$  by less than  $\epsilon > 0$ . Then if  $A = (Q, C, TO, M, q_{in}, VS')$  is a control automaton with a finite set of control laws  $C$  as its output alphabet which is correct for the viability set  $VS$ , then the closure of  $A$ ,  $\overline{A}$ , is also correct for the viability set  $VS$ .*

Lastly the following is a result on the existence of a fixed point of the controllability operator.

**Theorem 2.29.** *Suppose that the plant is either modelled by a differential inclusion  $\dot{x} \in F(x, c)$  satisfying conditions (I) and (II) or is modelled by a vector differential equation  $\dot{x} = f(x, c, d)$  satisfying conditions (I\*) and (II\*). Assume that the set of plant state measurements  $M$  consists of all values which deviate from the actual plant states in the viability set  $VS$  by less than  $\epsilon > 0$ . Let  $VS$  be a viability set which is compact and let  $H$  denote the controllability operator for the plant relative to  $VS$  and some fixed length of a sampling interval  $\Delta$ . Suppose that the admissible control laws maintain a constant control law in any interval and that up to translation, there are only a finite number of control laws  $C$ . Let  $Z_0, Z_1, \dots$  be defined inductively by*

$$\begin{aligned} Z_0 &= VS \\ Z_{n+1} &= H(Z_n), \quad n \geq 0 \end{aligned}$$

and let  $V = \bigcap_{n \geq 0} Z_n$ . Then

1. If  $Z_n$  is empty for some  $n$ , the controllability operator  $H$  of the plant has no nonempty fixed points.
2. If  $Z_n$  is nonempty for all  $n$ , then  $V$  is a maximal fixed point of the controllability operator  $H$  which is nonempty.

## 2.2 Aubin *et al* Approach to Viability of Hybrid Systems [15]

In [15] impulse differential inclusions are introduced to model hybrid phenomenon. In [15] both viability and invariance properties are examined. Below, only viability of impulse differential inclusions will be considered.

In order to introduce an order on the times between discrete transitions the following defines the notion of a hybrid time trajectory.

**Definition 2.30.** *A hybrid time trajectory  $\tau = I_{i=0}^N$  is a finite or infinite sequence of intervals of the real line, such that*

- for  $i < N$ ,  $I_i = [\tau_i, \tau'_i]$ ;
- if  $N < \infty$ , then either  $I_N = [\tau_N, \tau'_N]$ , or  $I_N = [\tau_N, \tau'_N[$ , possibly with  $\tau'_N = \infty$ ;
- for all  $i$ ,  $\tau_i \leq \tau'_i = \tau_{i+1}$ .

Only time invariant systems will be considered so it can be assumed that  $\tau_0 = 0$ .

An impulse differential inclusion is defined to capture hybrid phenomenon. It can be considered as a variant of a hybrid automata.

**Definition 2.31.** *An impulse differential inclusion is a collection  $H = (X, F, R, J)$ , consisting of a finite-dimensional vector space  $X$ , a set valued map  $F : X \rightarrow 2^X$ , regarded as a differential inclusion  $\dot{x} \in F(x)$ , a set valued map  $R : X \rightarrow 2^X$ , regarded as a reset map, and a set  $J \subseteq X$ , regarded as a forced transition set.*

The state of the impulse differential inclusion is  $x \in X$ .

**Definition 2.32.** *A run of an impulse differential inclusion,  $H = (X, F, R, J)$ , is a pair  $(\tau, x)$ , consisting of a hybrid time trajectory  $\tau$  and a map  $x : \tau \rightarrow X$ , that satisfies*

- Discrete Evolution: for all  $i$ ,  $x(\tau_{i+1}) \in R(x(\tau'_i))$ ;
- Continuous Evolution: if  $\tau_i < \tau'_i$ ,  $x(\cdot)$  is a solution to the differential inclusion  $\dot{x} \in F(x)$  over the interval  $[\tau_i, \tau'_i]$  starting at  $x(\tau_i)$ , with  $x(t) \notin J$  for all  $t \in [\tau_i, \tau'_i[$ .

A run of a differential inclusion begins with some  $x_0 \in X$  under the dynamics  $\dot{x} = F(x)$  and continues until  $x \in J$  at which time a discrete jump is taken. For states where  $R(x) \neq \emptyset$  a discrete jump from  $x$  to some value in  $R(x)$  can be taken. Therefore,  $R$  enables transitions while the set  $J$  forces transitions. In some instances the system may *block* which occurs when  $x \in J$  but  $R(x) = \emptyset$ .

The notion of continuous-time viability is adopted for hybrid systems.

**Definition 2.33.** *A run  $(\tau, x)$  of an impulse differential inclusion,  $H = (X, F, R, J)$ , is called viable in a set  $K \subseteq X$  if for all  $t \in \tau$ ,  $x(t) \in K$ .*

An infinite run is a run  $(\tau, x)$  where either  $\tau$  is an infinite sequence or  $\sum_i (\tau'_i - \tau_i) = \infty$ .  $\mathcal{R}_H(x_0)$  denotes the set of all runs of  $H$  and  $\mathcal{R}_H^\infty(x_0)$  denotes the set of infinite runs of  $H$ .

**Definition 2.34.** A set  $K \subseteq X$  is called viable under an impulse differential inclusion,  $H = (X, F, R, J)$ , if for all  $x_0 \in K$  there exists an infinite run,  $(\tau, x) \in \mathcal{R}_H^\infty(x_0)$ , viable in  $K$ .

As with the notion of viability, the viability kernel from continuous-time systems is adopted for hybrid systems.

**Definition 2.35.** The viability kernel  $Viab_H(K)$  of a set  $K \subseteq X$  under an impulse differential inclusion,  $H = (X, F, R, J)$ , is the set of states  $x_0 \in X$  for which there exists an infinite run,  $(\tau, x) \in \mathcal{R}_H^\infty(x_0)$ , viable in  $K$ .

A map  $F : X \rightarrow 2^X$  is *Marchaud* if and only if

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

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

The *contingent cone* to a set  $K$  at a point  $x \in X$  is denoted by  $T_K(x)$ .

**Lemma 2.36.** Consider a Marchaud map  $F : X \rightarrow 2^X$  and two closed sets  $K \subseteq X$  and  $C \subseteq X$ . For all  $x_0 \in K$ , there exists a solution of  $\dot{x} \in F(x)$  starting at  $x_0$  which is either

1. defined over  $[0, \infty[$  with  $x(t) \in K$  for all  $t \geq 0$ ;
2. defined over  $[0, T]$  for some  $T \geq 0$ , with  $x(T) \in C$  and  $x(t) \in K$  for all  $t \in [0, T]$ ;

if and only if for all  $x \in K \setminus C$ ,  $F(x) \cap T_K(x) \neq \emptyset$ .

Theorems on viability of an impulse differential inclusion depend on whether  $J$  is open or closed. The result for  $J$  closed is given by the following.



**Theorem 2.37.** *Consider an impulse differential inclusion  $H = (X, F, R, J)$  such that  $F$  is Marchaud,  $R$  is upper semicontinuous with closed domain and  $J$  is closed. A closed set  $K \subseteq X$  is viable under  $H$  if and only if*

1.  $K \cap J \subseteq R^{-1}(K)$ ;
2.  $\forall x \in K \setminus R^{-1}(K), F(x) \cap T_K(x) \neq \emptyset$ .

The result on viability for  $J$  open is given by the following.

**Theorem 2.38.** *Consider an impulse differential inclusion  $H = (X, F, R, J)$  such that  $F$  is Marchaud,  $R$  is upper semicontinuous with closed domain and  $J$  is open. A closed set  $K \subseteq X$  is viable under  $H$  if and only if*

1.  $K \cap J \subseteq R^{-1}(K)$ , and
2.  $\forall x \in (K \cap I) \setminus R^{-1}(K), F(x) \cap T_{K \cap I}(x) \neq \emptyset$ .

*Assumption 1:* An impulse differential inclusion  $(X, F, R, J)$  is said to satisfy *Assumption 1* if  $J \subseteq R^{-1}(X)$  and, if  $J$  is open (hence  $I = X \setminus J$  is closed),  $F(x) \cap T_I(x) \neq \emptyset$ , for all  $x \in I \setminus R^{-1}(X)$ .

A finite run  $(\tau, x)$  is one for which  $\tau$  is a finite sequence ending with a compact interval.

**Corollary 2.39.** *Consider an impulse differential inclusion  $H = (X, F, R, J)$  such that  $F$  is Marchaud, and  $R$  is upper semicontinuous with closed domain and  $J$  is either open or closed. Every finite run of  $H$  can be extended to an infinite run if and only if  $H$  satisfies Assumption 1.*

For a differential inclusion  $\dot{x} \in F(x)$ , the viability kernel of a set  $K$  with target  $C$ , denoted by  $Viab_F(K, C)$ , is defined as the set of states for which there exists a solution to the differential inclusion that remains in  $K$  either forever or until it reaches  $C$ .

**Lemma 2.40.** *Consider a Marchaud map  $F : X \rightarrow 2^X$  and two closed subsets of  $X$ ,  $K$  and  $C$ .  $Viab_F(K, C)$  is the largest closed subset of  $K$  satisfying the conditions of Lemma 2.36.*

To provide a fixed point characterization of viability, consider the operator  $Pre_H^\exists : 2^X \rightarrow 2^X$  defined by

$$Pre_H^\exists(K) = Viab_F(K \cap I, R^{-1}(K)) \cup (K \cap R^{-1}(K)).$$

where  $I = X \setminus J$ .

**Lemma 2.41.** *Consider an impulse differential inclusion  $H = (X, F, R, J)$  such that  $F$  is Marchaud,  $R$  is upper semicontinuous with closed domain, and  $J$  is open. A closed set  $K \subseteq X$  is viable under  $H$  if and only if it is a fixed point of the operator  $Pre_H^\exists$ .*

**Theorem 2.42.** *Consider an impulse differential inclusion  $H = (X, F, R, J)$  such that  $F$  is Marchaud,  $R$  is upper semicontinuous with closed domain and compact images, and  $J$  is open. The viability kernel of a closed set  $K \subseteq X$  under  $H$  is the largest closed subset of  $K$  viable under  $H$ , that is, the largest closed fixed point of  $Pre_H^\exists$  contained in  $K$ .*

The proof of the above theorem depends on the following algorithm.

**Algorithm 1 (Viability Kernel Approximation)**

initialization:  $K_0 = K, i = 0$

repeat

$$K_{i+1} = Pre_H^\exists(K_i)$$

$$i = i + 1$$

until  $K_i = K_{i+1}$

In the case that the Viability Kernel Approximation Algorithm does not converge, the following result indicates that the algorithm provides successively better approximations to the viability kernel.

**Lemma 2.43.** *Consider an impulse differential inclusion  $H = (X, F, R, J)$  such that  $F$  is Marchaud,  $R$  is upper semicontinuous with closed domain and  $J$  is open. Let  $K \subseteq X$  be a closed set and  $K_i$  be the sequence of sets generated by the Viability Kernel Approximation Algorithm. Then  $x_0 \in K_N$  if and only if there exists a run  $(\tau, x) \in \mathcal{R}_H(x_0)$  that remains in  $K$  for at least  $N$  jumps.*

## 2.3 Deshpande–Varaiya Approach to Viability of Hybrid Systems [34]

In [34], hybrid dynamics are captured by what is referred to as a *hybrid transition system*. This hybrid transition system is specified by the tuple

$$H = (Q, \mathbb{R}^n, \Sigma, E, \Phi)$$

where

- $Q$  is the finite set of discrete states.
- $\mathbb{R}^n$  is the set of continuous states.
- $\Sigma$  is the finite set of discrete events.
- $E$  is the finite set of edges

$$E \subset Q \times \mathcal{P}(\mathbb{R}^n) \times \Sigma \times \mathbb{R}^n \rightarrow \mathbb{R}^n \times Q$$

with an edge  $e \in E$  denoted by

$$(q_e, X_e, V_e, r_e, q'_e)$$

and enabled when the discrete state is  $q_e$  and the continuous state is in  $X_e$ . The event  $V_e \in \Sigma$  is accepted by the system when the event  $e$  is taken. The continuous state is reset according to  $r_e$  and the system enters  $q'_e$ .

- $\Phi$  is a set of differential inclusions modelling the continuous dynamics

$$\Phi = \{F_q : \mathbb{R}^n \rightarrow \mathcal{P}(\mathbb{R}^n) \setminus \emptyset \mid q \in Q\}$$

In discrete state  $q$ , the continuous dynamics evolve according to the differential inclusion

$$\dot{x}_c(t) = F_q(x_c(t))$$

The system evolves in phases, with the continuous state evolving over time until a discrete transition occurs at which time an instantaneous transition occurs. Let

$$\tau = [\tau'_0, \tau_1], [\tau'_1, \tau_2], [\tau'_2, \tau_3], \dots, \tau'_0 = 0 \text{ and } (\tau_i = \tau'_i \text{ and } \tau_{i+1} \geq \tau_i)$$

be a sequence of intervals of  $\mathbb{R}_+$ . Let  $T$  be the set of all such interval sequences. For each  $i$ ,  $H$  evolves continuously on  $(\tau'_i, \tau_{i+1})$  and makes a discrete transition at  $\tau_{i+1}$ . If  $\tau'_i = \tau_{i+1}$  then the system makes a discrete transition without any continuous evolution.

The semantics of the hybrid transition system are given over traces  $s = (x_d, x_c, v)$  where  $x_d$  is the discrete state trace,  $x_c$  is the continuous state trace and  $v$  is the edge transition trace. Let the set of all traces be

$$S = \{(\tau \rightarrow Q, \tau \rightarrow \mathbb{R}^n, \tau \rightarrow E \cup \{null\}) | \tau \in T\},$$

where *null* denotes the absence of a transition.  $s \in S$  given over  $\tau$  is a run of  $H$  if the following two conditions hold.

1. Continuous evolution.

For each  $i$

- a)  $\forall t \in (\tau'_i, \tau_{i+1})(v(t) = null)$ ,
- b)  $x_d(\cdot)$  is constant over the interval  $[\tau'_i, \tau_{i+1}]$
- c)  $\forall t \in (\tau'_i, \tau_{i+1})(\dot{x}_c(t) \in F_{x_d(t)}(x_c(t)))$

2. Discrete evolution.

At each boundary point  $\tau_i = \tau'_i$

- a)  $v(\tau_i)$  is an edge  $e \in E$ ,
- b)  $x_d(\tau_i) = q_e$  and  $x_c(\tau_i) \in X_e$ , i.e., the edge  $e$  is enabled at  $\tau_i$ ,
- c)  $x_d(\tau'_i) = q'_e$  and  $x_c(\tau'_i) = r_e(x_c(\tau_i))$ , i.e., the state is reset instantaneously.

The set of all runs that have infinitely many transitions referred to as  $\omega$ -runs of  $H$  is

$$S_H = \{s \text{ run of } H | s \text{ is defined over an infinite sequence } \tau \in T\}$$

For  $F_q \in \Phi$ , let the state evolution map be  $\phi_q : \mathbb{R}_+ \times \mathbb{R}^n \rightarrow \mathcal{P}(\mathbb{R}^n)$  which is the set of continuous states that can be reached from  $x_0$  at time  $t$  under the inclusion

$$\dot{x}_c(t) \in F_q(x_c(t)), \quad x_c(0) = x_0.$$

**Definition 2.44.** *The forward projection of  $x_0$  under  $\phi_q$  is given by*

$$\phi_q(x_0) = \cup_{t \geq 0} \phi_q(t, x_0)$$

**Definition 2.45.** *The backward projection of  $x \in \mathbb{R}^n$  under  $\phi_q$  is given by*

$$\beta_q(x) = \{x_0 | x \in \phi_q(x_0)\}$$

and for  $X \subset \mathbb{R}^n$

$$\beta_q(X) = \cup_{x \in X} \beta_q(x)$$

**Definition 2.46.** For  $e \in E$  the nonblocking switch-and-coast operator  $n_e : \mathcal{P}(\mathbb{R}^n) \rightarrow \mathcal{P}(\mathbb{R}^n)$  is given by

$$n_e(Y) = X_e \cap r_e^{-1}(\beta_{q'_e}(Y))$$

**Lemma 2.47.**  $n_e$  distributes over  $\cup$ , i.e.,  $n_e(Y_1 \cup Y_2) = n_e(Y_1) \cup n_e(Y_2)$ .

**Definition 2.48.** For  $q \in Q$  define  $out(q)$  as the edges (including self-loops) that exit  $q$ ,

$$out(q) = \{e \in E | q_e = q\}$$

Let  $E$  be ordered as  $\{e_1, \dots, e_{|E|}\}$  and let

$$X_H = \{X_{e_1}, \dots, X_{e_{|E|}}\}^T \in [\mathcal{P}(\mathbb{R}^n)]^{|E|}$$

be the vector of edge enabling conditions.

**Definition 2.49.**  $N : [\mathcal{P}(\mathbb{R}^n)]^{|E|} \rightarrow \mathcal{P}(\mathbb{R}^n)^{|E|}$  is defined as

$$[N(Z)]_i = n_{e_i}(\cup_{j|e_j \in out(q'_e)} Z_j).$$

Using Lemma 2.47,  $[N(Z)]_i$  can be written as

$$[N(Z)]_i = \cup_{j|e_j \in out(q'_e)} n_{e_i}(Z_j)$$

**Definition 2.50.** An edge sequence  $e_1, \dots, e_l$  is said to be a path if  $\forall i = 1, \dots, l-1 (q'_{e_i} = q_{e_{i+1}})$ . It is said to be a loop if it is a path and  $q_{e_1} = q'_{e_l}$ . It is a simple loop if it is a loop and  $\forall i, j = 1, \dots, l (i \neq j \Rightarrow q_{e_i} \neq q_{e_j})$ .

**Definition 2.51.** Define the viability kernel of  $H$  as  $X^* = \cap_k N^k(X_H)$ .

The meaning of the viability kernel can be taken as ensuring the existence of an infinite run starting at  $v(0) = e$  and  $x_c(0) = x$  for any  $x \in X_e^*$  and any  $e$ .

**Definition 2.52.** For  $q \in Q$  define  $in(q)$  as the edges that enter  $q$ ,

$$in(q) = \{e | q'_e = q\}$$

Also, the following two sets will be needed to establish existence of the viability kernel:

$$X_q^{out} = \cup_{e \in out(q)} X_e \text{ and } X_q^{in} = \cup_{e \in in(q)} r_e(X_e)$$

**Theorem 2.53.** *Assume that*

1. *for each  $e \in E$ ,  $X_e$  is compact and  $r_e$  is continuous, and*
2. *for each  $q \in Q$ , the set  $\{(x, x') | x \in X_q^{in}, x' \in X_q^{out} \cap \phi_q(x)\}$  is closed.*

*Then  $N(X^*) = X^*$ .*

Next, conditions are given for there to exist some  $K$  such that

$$N^K(X_H) = X^*$$

**Theorem 2.54.** *Suppose each simple loop  $e_1, \dots, e_l$  of  $H$  has an edge  $e_1$  such that*

1.  *$r_{e_1}(X_{e_1})$  is bounded, and*
2.  *$\exists \rho_{e_1} \forall x \in X_{e_2} \forall y \in \beta_{q'_{e_1}} [\beta_{q'_{e_1}} \cap \phi_{e'_1}(y) \text{ is a union of } \rho_{e_1} \text{-balls}]$ .*

*Then  $\exists K(N^K(X_H) = X^*)$ .*

Another result on the computability of the viability kernel is given by the following theorem.

**Theorem 2.55.** *Suppose each simple loop  $e_1, \dots, e_l$  of  $H$  has an edge  $e_1$  such that either  $e_1$  satisfies the conditions of Theorem 2.54 or*

1.  *$X_{e_1}$  is bounded, and*
2.  *$\exists \rho_{e_1} > 0 \forall y \in \beta_{q'_{e_1}}(X_{e_2}) (r_{e_1}^{-1}(y) = \emptyset \text{ or } r_{e_1}^{-1}(y) \text{ is a union of } \rho_{e_1} \text{ balls})$ .*

*Then  $\exists K(N^K(X_H) = X^*)$ .*

There are instances where the reset map  $r_e$  may need to be set-valued, denoted  $R_e$ . In this case where the controller can choose the value that the continuous state is reset to, the reset is said to be controlled and is defined by

$$R_e^{-1}(Y) = \{x | R_e(x) \cap Y \neq \emptyset\}$$

If the choice of reset value is not available to the controller then the reset map is defined by

$$R_e^{-1}(Y) = \{x \mid R_e(x) \subset Y\}$$

For set-valued reset maps, the following theorem applies.

**Theorem 2.56.** *If  $H$  satisfies the conditions of Theorem 2.55 and for all edges  $e$  with an uncontrolled set-valued reset map  $\text{out}(q'_e)$  is a singleton, then  $\exists K(N^K(X_H) = X^*)$ .*

From now on, it will be assumed that the differential inclusion arises from the control variables, i.e., for each  $q$

$$F_q(\cdot) = \{f_q(\cdot, w) \mid w \in \mathbb{R}^m\}$$

for some  $f_q : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ .

The set of edges  $E$  is taken to be composed of  $E_p$ ,  $E_f$  and  $E_u$  where  $E_p$  are prohibitable,  $E_f$  are forcible and  $E_u$  are uncontrollable. The edge subsets satisfy  $E_p \cap E_u = E_f \cap E_u = \emptyset$ . Let  $Y$  be the set of possible observations

$$Y = Q \times \mathbb{R}^n$$

with  $y : \tau \rightarrow Y$  an observation trace. Let  $S_Y$  be the set of all observation traces. Let  $U$  be the set of possible controls

$$U = \mathcal{P}(E_p) \times \mathcal{P}(E_f) \times \mathbb{R}^m$$

where a control trace is given by  $u = (u_p, u_f, u_c) : \tau \rightarrow U$  with  $u_p$  and  $u_f$  piecewise constant and  $u_c$  measurable control functions. Let  $S_U$  be the set of all control traces.

Available to the controller is a special forcible edge called *block* which is always enabled. Let the set of extended controls be  $U^B$

$$U^B = \mathcal{P}(E_p) \times [\mathcal{P}(E_f) \cup \{\text{block}\}] \times \mathbb{R}^m$$

and let  $S_{U^B}$  be the set of all extended control traces such that

$$\forall u \in S_{U^B} \forall t [u_f(t) = \{\text{block}\} \Rightarrow \forall t' > t (u_f(t') = \{\text{block}\})]$$

Let  $S \subset S^B$  be the set of all extended traces  $(x_d, x_c, v)$  where  $v$  is allowed to be *block* or *null*.

Given an observation trace  $y = (x_d, x_x) \in S_Y$  the set of extended system traces that can generate  $y$  is given by the inverse projection

$$G_y = \{(x_d, x_c, v) \in S^B\}.$$

Given a control trace  $u = (u_p, u_f, u_c) \in S_{UB}$  the set of extended system traces that can be produced in response to  $u$  is given by

$$\begin{aligned} G_u = \{ & (x_d, x_c, v) \in S^B \mid \forall t [v(t) \notin u_p(t), \\ & \exists e \in u_f(t)(x_d(t) = q_e \text{ and } x_c(t) \in X_e) \\ & \Rightarrow \exists e \in u_f(t)(v(t) = e) \text{ and } v(t) \neq \text{block} \\ & \Rightarrow (\dot{x}_c(t) = f_{x_d(t)}(x_c(t), u_c(t)))]\} \end{aligned}$$

For a trace  $s$  let  $s_t$  denote the prefix of  $s$  up to  $t$  and for a prefix  $s_t$  let  $\text{ext}(s_t) = \{s' \mid s'_t = s_t\}$  denote the set of traces that extend  $s_t$ .

A control strategy is a point-to-set map

$$\kappa : S_Y \rightarrow \mathcal{P}(S_{U'}) \setminus \emptyset$$

The extended closed loop behaviour of  $\kappa$  is given by

$$G_\kappa = \{G_u \cap G_y \mid u \in \kappa(y)\}$$

and the closed-loop behaviour of  $\kappa$  is

$$S_\kappa = G_\kappa \cap S.$$

**Definition 2.57.** Fix  $\alpha > 0$  and a hybrid system  $H$ . A point-to-set map  $\kappa_H^\alpha : S_Y \rightarrow \mathcal{P}(S_{U'}) \setminus \emptyset$  is an  $\alpha$ -lag possibly blocking control strategy for  $H$  iff

1.  $S_{\kappa_H^\alpha} \subset S_H$ ,
2.  $\forall y, y' \in S_Y \forall t (y_{(t+\alpha)} = y'_{(t+\alpha)} \Rightarrow \{u_t \mid u \in \kappa_H^\alpha(y)\} = \{u_t \mid u \in \kappa_H^\alpha(y')\})$ , and
3.  $\forall y \forall u \forall t [\forall y' \in \text{ext}(y_{(t+\alpha)}) \forall u' \in \text{ext}(u_t)$

$$\begin{aligned} (G_u \cap G_{y'} \cap S_H = \emptyset) & \Rightarrow \forall u'' \in \text{ext}(u_t) \cap \kappa_H^\alpha(y) \\ & \forall \tau > t (u_f''(\tau) = \{\text{block}\})]. \end{aligned}$$



If for an  $\alpha$ -lag possibly nonblocking control strategy  $\kappa$ ,  $G_\kappa \subset S$  then  $\kappa$  is called an  $\alpha$ -lag nonblocking control strategy for  $H$ , denoted as  $\tilde{\kappa}_H^\alpha$ .

**Definition 2.58.** *The 0-lag nonblocking control strategy is given by*

$$\tilde{U}_H = \cap_{\alpha > 0} \tilde{U}_H^\alpha$$

Let the state evolution map be denoted by  $\phi_q^u$  for controlled vector field  $f_q(x(t), u(t))$  and measurable control law  $u(t)$ .

The hybrid controller is given as a (partial) map

$$\gamma_H = (\gamma_p, \gamma_f, \gamma_c) : Q \times \mathbb{R}^n \rightarrow \mathcal{P}(E_p) \times \mathcal{P}(E_f) \times [\mathcal{P}(\{\mathbb{R}_+ \rightarrow \mathbb{R}^m\}) \setminus \{\emptyset\}]$$

where

$$\begin{aligned} \gamma_p(q, x) &= \{e \in E_p \mid e \in \text{out}(q) \text{ and } x \notin X_e^*\}, \\ \gamma_f(q, x) &= \{e \in E_f \mid e \in \text{out}(q), x \in X_e^*, \\ &\quad \text{and } \forall e' \in \text{out}(q) [F_q(x) \cap T_{\beta_q}(X_{e'}^*)(x) = \emptyset]\}, \\ \gamma_c(q, x) &= \{u \mid u \text{ measurable and } \exists e \in \text{out}(q) \exists t \geq 0 (\phi_q^u(t, x) \in X_e^*)\} \end{aligned}$$

**Definition 2.59.**  *$H$  is said to be viable iff*

1.  $\exists e \in E(X_e^* \neq \emptyset), \forall e \in E(X_e^* \text{ is closed}), N(X^*) = X^*$ , and  $\forall e \in E(e \notin E_p \cap E_f \Rightarrow X_e^* = X_e)$ , and
2. in any state  $(q, x)$ , at any time  $t$ , given  $u_p(t)$  if there exists an outgoing edge  $e \notin u_p(t)$  such that

$$x \in X_e, F_q(x) \cap T_{\beta_q(X_e)}(x) = \emptyset$$

and

$$\forall e' \neq e (e' \in \text{out}(q) \Rightarrow e' \in u_p(t) \text{ or } x \notin \beta_q(X_{e'}))$$

then  $e$  is taken at  $t$ .

**Theorem 2.60.** *If  $H$  is viable then*

$$\{(x_d, x_c, v) \in G_{\gamma_H} \mid x_c(0) \in \beta_{x_d(0)}(\cup_{e \in \text{out}(x_d(0))} X_e^*)\} = S_{\tilde{U}_H}$$

## 2.4 Related Literature

In this section, we review existing literature on hybrid systems that is related to this work. This is carried out by first considering the overall area of hybrid systems. Literature related specifically to Chapters 3–8 is then reviewed.

### HYBRID SYSTEMS

The collections appearing as *Lecture Notes in Computer Science* series provide comprehensive coverage of hybrid systems research, see for example, [44], [4], [3], [89], and the special issues [6], [5]. Modeling, control, and analysis of hybrid systems are reviewed in [20], [19], [53]. A number of theses on hybrid systems have been completed. We mention in particular those of Deshpande [33] and Branicky [18] as providing interesting results and discussion on a variety of aspects arising in hybrid systems, and in particular their control.

### CHAPTER 3 HYBRID MODEL

A useful distinction between hybrid models is given in [19] as *automatization* where component systems of the hybrid system are treated as automata and *systemization* where component systems are treated as dynamical systems. Component systems existing in both continuous and discrete domains and interacting are basic hybrid attributes that each of these two classes of approaches address. Nerode and Kohn, through their *Topological Hybrid System (THS)* [68] provide a specific model instance for the automatization class, while Branicky, through his *Controlled General Hybrid Dynamical System (CGHDS)* provides a specific model instance for the systemization class. The *Hybrid Transition System (HTS)* [33] provides a model that could be said to lie ‘in between’ these two classes; a transition system model being a generalization of an automaton and a specialization of a dynamical system.

We consider the above three models relative to five characteristics that we note in Chapter 3 as being desirable in a hybrid model. Both THS and HTS describe single-valued and multivalued constituent dynamics, whereas a CGHDS describes single-valued dynamics. Transition dynamics are captured in the CGHDS model by a jump delay mapping defining the delay between when an

uncontrolled jump in state is said to have occurred and when it actually occurs. Parametric uncertainty can be addressed using THS and HTS models by using multivalued dynamic descriptions. Structural uncertainty as considered in this work is not explicitly a part of these models. The THS and HTS allow controlled and uncontrolled transitions to depend on both state and time. The CGHDS model gives transitions which are explicitly dependent on the state. The THS includes sampling, whereas the HTS and CGHDS models and assumes that continuous plant state information is available. Related to transition dynamics, Nerode and Kohn [71] provide a model of plants with delay in switching between control parameter values. Incorporating dynamic specifications in a typical hybrid closed-loop and the affect on the A/D map is examined in [31]. The work of [94] provides a model for and a stability analysis of the timing delay between the generation of a discrete-event signal and the generation of the corresponding signal as an input to the plant. Within the context of a *hybrid automaton* [2], assignment of the data variables as part of the definition of the transition relation [2] or definition of the reset mapping [21] could be viewed as instances of transition dynamics models which are defined on the state variables for a hybrid automaton.

The study of the Nerode and colleague approach to hybrid systems applied to the three fluid-filled tank example presented in this chapter was first investigated in [60].

#### CHAPTER 4 VIABILITY

Before turning to viability and hybrid systems, we will briefly consider continuous-time viability. Continuous-time viability is considered in the monographs [11], [12] and [13], [7]. Necessary and sufficient conditions for the time-dependent viability problem are given in [37], [61]. A number of qualitative properties of linear dynamical inclusions, with a specific aim being to consider questions of approximation and computation is examined in [51]. Results based on nonsmooth analysis of continuous-time (weak) invariance (or viability) are given in [27], [28], including discussion of similarities and differences between the conditions arising based on a viability versus nonsmooth analytic approach.

Two approaches to viability of hybrid systems are given by the work of Nerode and colleagues and Varaiya and colleagues; we will denote these by  $\text{Viability}(\text{Nerode})$  and  $\text{Viability}(\text{Varaiya})$  respectively. First, we consider the  $\text{Viability}(\text{Nerode})$ . Two related approaches to viability are given in [49] and [70]. More general results are given in [70] for specific cases. In [49], *abstract local viability graphs* are used to provide conditions under which finite control automata can be found to ensure that the plant states remain within a viability set when there is no uncertainty in state and/or control. Extensions to this viability graph are introduced as a means to account for both state and control uncertainty. These extensions are defined over the graph states which characterize subsets of the continuous state space. In [70], the viability problem is posed as having to find a subset  $VS'$  of a subset of plant states  $VS$  and a control strategy such that, if the plant state begins in  $VS'$ , then the plant trajectories under the control strategy remains in  $VS$  forever. This approach is carried based on the definition of the controllability operator requiring that two conditions be satisfied over the sampling interval. A fixed point of the controllability operator along with an appropriate partitioning of the fixed point provides a design basis for a finite control automaton that ensures viability.

The  $\text{Viability}(\text{Varaiya})$  approach, initiated by [33], defines viability relative to the Hybrid Transition System (HTS), and ensures that an infinite number of transitions are possible. Necessary conditions are given for the existence of the viability kernel in general as well as for its existence in a finite number of iterations for three cases for which an HTS exhibits different structural properties. The viability property is shown to ensure the existence of a control strategy generating the largest nonblocking control strategy for the HTS. Under the assumption of a piecewise constant inclusion approximation to a differential inclusion, it is proved in [73] that a robustly invariant set (where robust is taken to mean ensuring the desired property of invariance in the presence of perturbations in the righthand side of the differential inclusion) can be computed from a finite graph approximation of the system trajectories. In [50], the viability kernel for an HTS under convex,

compact multivalued dynamics using an iterative algorithm which is stated to terminate in a finite number of steps.

A basic difference between these approaches is the definition of viability. Viability(Nerode) views viability and the viability problem as requiring constraints on the plant states to be satisfied for all time, analogously to the continuous-time case. The notion of viability developed by Viability(Varaiya) is based on conditions being given to ensure that some transition can always occur. This notion is dependent on the hybrid model. Another difference is that Viability(Nerode) includes sampling and plant measurement error as part of the problem setup while neither of these are included in Viability(Varaiya). In both approaches, resolution of computational issues is lacking. An appealing feature of Viability(Nerode) is that the problem of design or verification is based on an algorithmic solution that generates a control solution which explicitly takes into account the sampling interval, the system dynamics and the available control action.

Some other investigations on viability and related issues have been completed. The non-sampling viability problem was examined in the pioneering work of Aubin and co-workers [15]. In [45], what is referred to as an elementary hybrid machine is taken as the hybrid model that is used to provide a control synthesis technique for ensuring that the controlled system is legal and minimally restrictive. The synthesis is carried out based on what is referred to as bounded-rate hybrid machines, the main restriction of this is that the dynamics in a given vertex are only assumed to be known to lie within an interval defined by constant upper and lower bounds. A run (sequence through successive vertices) is referred to as *viable* or non-Zeno if an infinite number of transitions cannot occur in finite time. In [86], a hybrid automaton model is adopted with a conceptual algorithm given for finding the maximal controlled invariant set; solution of a Hamilton-Jacobi equation is proposed as a computational tool for computing the maximally invariant set and for generating the least restrictive controller ensuring invariance as many inputs and as many transitions as possible are allowed at each state that force the smallest number of transitions. Designing controllers to meet safety specifications is

examined in [40], [41] while [90] provide an overview of hybrid system verification based on game theory. Homotopies defined over the set of trajectory solutions arising from hybrid automata is given in [21] for rectangular differential inclusions and more general Lipschitz inclusions. Based on a partition of state space, a lattice is shown to exist with component element machines defined by the partitioning.

The fixed point of the controllability operator can be taken as the sampled version of the continuous-time viability kernel [30], [38], [81], [75], [9], [35]. Approximating the fixed point of the controllability operator is examined in [56], [57]. An algorithm for construction of the hybrid equivalent of the viability kernel referred to as the hybrid kernel is given in [82]. Existence of solutions and viability for systems modelled by differential inclusions subject to impulsive set-valued state resets is examined in [64]. Lastly, alternative approaches to viability of hybrid systems are given in [32], [46], [80], [87].

## CHAPTER 5 ROBUST VIABILITY

The work on robust viability extends the development of the controllability operator approach to viability given in [70], [56], [52]. The problem of robust viability has not yet been explored extensively in the literature with a preliminary investigation given in [58].

## CHAPTER 6 VIABILITY IN PRACTICE

The Active Magnetic Bearing example was studied in [22]. The work presented in Chapter 6 on viable cascade control of a batch polymerization problem is taken from [54], [55].

## CHAPTER 7 AN OPERATOR APPROACH TO VIABLE ATTAINABILITY OF HYBRID SYSTEMS

The problem of attainability has been treated in the literature as a target problem in [74], [76]. Viable attainability has been considered in the literature as characterization of what is referred to as the capture basin. This characterization has been performed for differential inclusions in [10], [8], [9] and for impulse control systems in [82], [14]. The Reach-Avoid operator in [88] can be used to characterize the viable attainability problem. The work presented in Chapter 7 on viable attainability is taken from [59].

## CHAPTER 8 SOME TOPICS RELATED TO THE CONTROLLABILITY OPERATOR

The work presented in this chapter is original material.

### 2.5 Conclusion

This chapter has revealed that the literature on hybrid systems and specifically viability of hybrid systems is rather rich and diverse. Several approaches to viability of hybrid systems were reviewed. The differences in approaches can be attributed to basic underlying differences in the tools applied to the problem of viability, including researchers from different disciplines tackling the problem.

Viability of Hybrid Systems  
A Controllability Operator Approach

Labinaz, G.; Guay, M.

2012, X, 246 p., Hardcover

ISBN: 978-94-007-2520-1