Solving Diagnosability of Hybrid Systems via Abstraction and Discrete Event Techniques

Alban.Grastien[1] and **Louise Travé-Massuyès**[2] and **Vicenç Puig**[3]
[1]Data61, Decision Science Program; and the ANU, Artificial Intelligence Group
e-mail: alban.grastien@data61.csiro.au
[2]LAAS-CNRS, Université de Toulouse, France
e-mail: louise@laas.fr
[3]SAC Group, Universitat Politècnica de Catalunya,Barcelona,Spain
e-mail: vicenc.puig@upc.edu

Abstract

This paper addresses the problem of determining the diagnosability of hybrid systems by abstracting hybrid models to a discrete-event setting. From the continuous model, the abstraction only remembers two pieces of information: discernability between modes (when they are guaranteed to generate different observations) and ephemerality (when the system cannot stay forever in a given set of modes). Then, we use standard DES diagnosability algorithms. The second contribution is an iterative approach to diagnosability that starts by generating the most abstract DES model of the hybrid system. This allows to check the diagnosability of this DES model. If it is diagnosable, that means we have found an abstraction of the hybrid model that is diagnosable. If it is not, the counterexample generated by the diagnosability procedure is analysed to refine the DES. If no refinement is found, then it can not be proved that the hybrid system is diagnosable. Otherwise, the refinement is included in the abstract DES model and diagnosability procedure continues.

1 Introduction

Diagnosability is the property of a system and its instrumentation guaranteeing that all anticipated faulty situations can be detected and identified without ambiguity on a bounded time window from the available observations of the system [Bayoudh and Travé-Massuyès, 2014].

Diagnosability has been studied for continuous systems and for discrete-event systems (DES) separately. In case of continuous systems, it is formulated in terms of fault detectability and isolability from a structural point of view as in [Blanke and Kinnaert, 2016] or accounting for the characteristics of model uncertainties and noises impacting the system [Basseville *et al.*, 2001]. In the case of DES, the first diagnosability definition was proposed in [Sampath *et al.*, 1995] together with the necessary and sufficient conditions for diagnosability based on the Sampath's diagnoser, a finite state machine built from the system model.

Diagnosability of hybrid systems was addressed later, benefiting from the works existing in both the continuous systems and the DES fields. [Bayoudh and Travé-Massuyès, 2014] exemplifies how these works can be merged for hybrid systems represented by hybrid automata. The discrete states of the hybrid automaton represent the operation modes of the system for which different continuous dynamics are specified via a set of differential equations involving continuous variables. The diagnosability of the continuously-valued part of the model is first analyzed and the new concept of mode signature is shown to characterize mode diagnosability from continuous measurements, also known as *discernibility*. Different mode signatures are then translated into a set of signature-events associated to mode transitions. The behavior of the hybrid system is hence modeled by a prefix-closed language over the original event alphabet enriched by these additional events. Based on this language, diagnosability analysis of the hybrid system is cast in a discrete-event framework. Other related works can be mentioned. For instance, the approach of [Daigle *et al.*, 2008] is similar to [Bayoudh and Travé-Massuyès, 2014] and [Cocquempot *et al.*, 2004] bases the analysis on continuous dynamics only and is hence limited to discernibility. [Vento *et al.*, 2015] extends the work of [Bayoudh and Travé-Massuyès, 2014] by proposing an incremental diagnosis framework in which discernibility remains implicit.

This paper addresses the problem of determining the diagnosability of hybrid systems with a different point of view. Instead of enriching the DES with full information arising from continuous dynamics (e.g. signature-events that require to determine all mode signatures as in [Bayoudh and Travé-Massuyès, 2014]), it proposes to abstract hybrid models H to a discrete-event setting D_M^0 and check diagnosability in an incremental way. The proposed approach starts by generating the most abstract DES model of the hybrid system model. Then, diagnosability of this DES model is checked. If it is diagnosable, that means we have found an abstraction of M that is diagnosable.

While if it is not, a counter example generated by the diagnosability procedure is generated searching for refinement that contradicts the counterexample. If no refinement is found, then it can not be proved that the hybrid system is diagnosable. Otherwise, the refinement is included in the abstract DES model and diagnosability procedure continues. This approach uses just the necessary information about continuous dynamics, in an "on request" manner, hence potentially saving quite a lot of computation.

The structure of the paper is as follows: Section 2 motivates and illustrates the approach with an easy to understand example. Section 3 provides the preliminaries regarding diagnosability. Section 4 describes the hybrid systems that we consider and the DES setting used for diagnosability. Section 5 shows how a hybrid system can be abstracted to a DES and the properties involved in refinment. Section 6 presents how to test diagnosability of a hybrid system incrementally, which is illustrated with the application example of Section 2. Section 7 draws the conclusions and indicates future research paths.

2 Application Example

We start with an example that illustrates our approach. We expect that little background knowledge is required to get an understanding of this section's content. The reader may prefer to read the next sections first and jump back here before the conclusion.

Consider the model of Figure 1. The system starts in mode $N1$ and navigates between $N1$, $N2$, and $N3$. With an initial value within $]0,90[$, the value of state variable x increases in $N1$ and $N2$, albeit at a different speed, and decreases in $N3$. Notice that the system can transition freely between $N1$ and $N2$ but that the system has to transition to $N3$ if the temperature becomes greater to 80. A fault leads to a similar situation where the increase/decrease in the state are modified and the models become uncertain. Observations are $y = x$ (the state), $\dot{y} = \dot{x}$ (the derivative of the state $\dot{x}$) and u, which takes a binary value.

Residuals and indicators – Residuals (cf. 5.1) are equations involving observable quantities that evaluate to zero in some modes. A set of primary residuals for this model can be generated as follows:

$$\begin{array}{l} r_{N1} = \dot{y} + y - 100 \\ r_{N2} = \dot{y} + y - 90 \\ r_{N3} = \dot{y} + y \\ r_{F1} = \dot{y} + y - [95,105] \\ r_{F2} = \dot{y} + y - [85,95] \\ r_{F3} = \dot{y} + y - [45,50] \end{array} \tag{1}$$

From these residuals, the set I of indicators introduced in Definition 2 can be obtained as explained in Section 5.1.

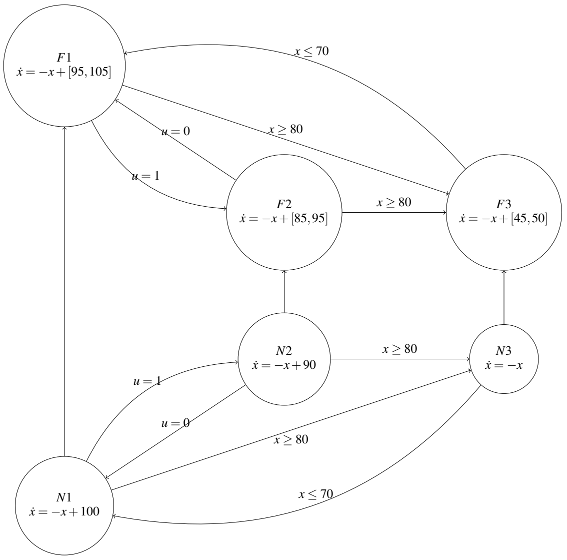

Figure 1: The System Model

Invariant sets – Taking into account the dynamic models associated to the modes of the hybrid system presented in Figure 1, the corresponding invariance sets can be computed as follows:

$$\begin{array}{l} \mathscr{S}_{N1} = 100 \\ \mathscr{S}_{N2} = 90 \\ \mathscr{S}_{N3} = 1 \\ \mathscr{S}_{F1} = [95,105] \\ \mathscr{S}_{F2} = [85,95] \\ \mathscr{S}_{F3} = [45,50] \end{array} \tag{2}$$

Analyzing the position of these invariant sets with respect to the mode's guards, we can assess ephemerability. Moreover, these invariant sets can also be used to evaluate the residuals associated to the set I of indicators that allow distinguishing in which mode q the hybrid system is operating. In particular, as e.g., residual r_{N1} will be zero and, consequently the associated indicator will be $I_{N1} = 1$ when the system is reaching the invariant set $\mathscr{S}_{N1}$ according to Section 5.1. Similar reasoning applies for residuals r_{N2} and r_{N3}. In case of residuals r_{F1} to r_{F3} may not evaluate to zero even if the hybrid system is in some of the associated modes because of the uncertainty, being the corresponding indicator $I_{N1} = -1$. Thus, invariant sets can be used in analogous manner as residuals to generate the indicators I introduced in Section 5.1.

Diagnosability analysis – The diagnosability procedure includes two components: a "discrete component", which generates "counter-examples" (that negate diagnosability) and a "continuous component", which tries to invalidate the counter-examples.

The discrete part is implemented by a twin plant method. It suffices to say that a counter-example is a pair of infinite behaviours, a faulty one and a nonfaulty one, that look similar. The semantics of the counter-example is that if (a prefix of) the faulty behaviour occurs, then the diagnostic engine might believe that the system is undergoing (a prefix of) the nominal behaviour.

The continuous component of the diagnosability procedure can debunk a counter-example either by proving that the infinite faulty behaviour is impossible ("ephemerality") or by proving that the pairs of modes involved in the two behaviours can always be distinguished.

The algorithm starts with an abstraction of the model where all the continuous aspects are ignored (loops are added on each mode). Each call to the continuous component refines the model.

Here are the details of the execution of the procedure on this example:

[CE1] The discrete component computes the following counter-example:

- If the system takes the following infinite faulty behaviour $b_f = N1 \to F1\ (\to F1)^{\infty}$,
- this behaviour cannot be distinguished from $b_N = N1 \to N1\ (\to N1)^{\infty}$.

This counter-example can be eliminated if we demonstrate that the infinite faulty behaviour is impossible (ephemerality, i.e., if the system cannot stay in mode $F1$ forever) or that the two behaviours can be distinguished (i.e., if $N1$ can always be distinguished from $F1$). In this instance, we see that $\{F1\}$ is ephemeral.

Proof of ephemerality is simply obtained from the invariant set of F_1: $\mathscr{S}_{F1} = [95, 105]$. For F_1, all the possible continuous state initial conditions are $x_0 = [0, 80[$, hence the system must cross the guard $x \geq 80$ to converge to the invariant set. The same result can be obtained by running hybrid reachability starting with $x_0 = [0, 80[$ as shown on Fig. 2.

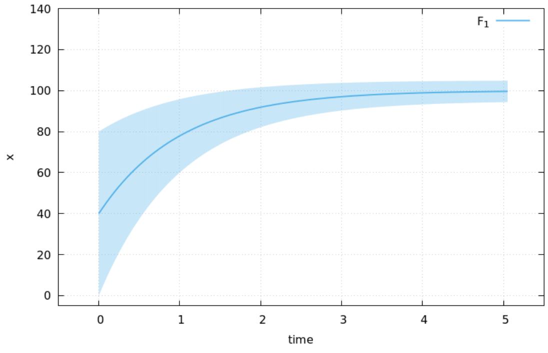

Figure 2: Reachability analysis for F1 starting with all possible initial states $x_0 = [0, 80[$

[CE2] The discrete component searches for a new counter-example. This counter-example is not allowed to include only the mode $F1$ in the loop of the infinite faulty behaviour. The new counter-example is

- $b_f = N1 \to F1 \to F2\ (\to F2)^{\infty}$,
- $b_N = N1 \to N1 \to N1\ (\to N1)^{\infty}$.

But $\{F2\}$ is ephemeral. Proof of ephemerality is the same as before: F_2: $\mathscr{S}_{F2} = [85, 95]$ and $x_0 = [0, 80[$, hence the system must cross the guard $x \geq 80$ to converge to the invariant set. Hybrid reachability would provide the same information.

[CE3] Now the discrete component is not allowed to generate a counter-example with a faulty loop that contains only $F1$ or only $F2$, but the faulty loop may consist in $\{F1, F2\}$. The new counter-example is

- $b_f = N1 \to F1 \to F2\ (\to F1 \to F2)^{\infty}$,
- $b_N = N1 \to N1 \to N1\ (\to N1 \to N1)^{\infty}$.

But $\{F1, F2\}$ is ephemeral. For a set of modes, proof of ephemerality is obtained by hybrid reachability. Possible initial states are $([0, 80[, F_1)$ and $([0, 80[, F_2)$. For any transition sequence triggered by $u(t)$, the system behavior converges towards the invariant set of the last mode. Fig. 3 and Fig. 4 show a 2 and 11 transitions scenario ending with mode F_2.

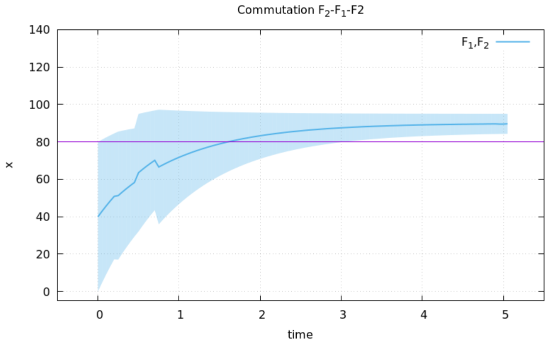

Figure 3: Reachability analysis for the set of modes {F1, F2} starting with all possible initial states $x_0 = [0, 80[$ and 2 transitions

[CE4] New counter-example:

- $b_f = N1 \to F1 \to F2 \to F3\ (\to F3)^{\infty}$,
- $b_N = N1 \to N1 \to N1 \to N1\ (\to N1)^{\infty}$.

But $\{F3\}$ is ephemeral. Proof of ephemerality is the same as before: $\mathscr{S}_{F3} = [45, 50]$ and $x_0 =]70, 80]$, hence the system must cross the guard $x \leq 70$ to converge to the invariant set.

[CE5] New counter-example:

- $b_f = N1 \to F1 \to F2 \to F3\ (\to F1 \to F2 \to F3)^{\infty}$,
- $b_N = N1 \to N1 \to N1 \to N1\ (\to N1 \to N1 \to N1)^{\infty}$.

The ephemerality does not allow to reject this counter-example. Therefore, we need to check whether $N1$ can always be distinguished from $F1$, whether $N1$ can always be distinguished from $F2$, and whether $N1$ can always be distinguished from $F3$. According to the invariant sets (2), $F2$ and $F3$ can be distinguished from $N1$ since the corresponding invariant sets do not intersect, but not $F1$ because in this case the intersection is not empty. This means that the residuals r_{F2} and r_{F3} can be used for distinguishing from $N1$. But this is not the case of residual r_{F1}.

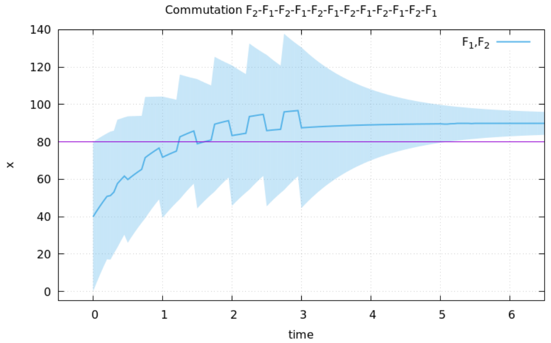

Figure 4: Reachability analysis for the set of modes {F1, F2} starting with all possible initial states $x_0 = [0, 80[$ and 11 transitions

[CE6] We generate a new counter-example that cannot have the faulty behaviour in mode $F2$ while the nominal behaviour is in mode $N1$:

- $b_f = N1 \to F1 \to F2 \to F3\ (\to F1 \to F2 \to F3)^\infty$,
- $b_N = N1 \to N1 \to N2 \to N1\ (\to N1 \to N2 \to N1)^\infty$.

But, $F3$ can be distinguished from $N1$ because the corresponding invariant sets do not intersect according to (2), or equivalently, the residuals r_{F3} and r_{N1} can be used for distinguishing between these two modes.

[CE7] New counter-example:

- $b_f = N1 \to F1 \to F2 \to F3\ (\to F1 \to F2 \to F3)^\infty$,
- $b_N = N1 \to N1 \to N2 \to N2\ (\to N1 \to N2 \to N2)^\infty$.

But, $F3$ can be distinguished from $N2$ because the corresponding invariant sets do not intersect according to (2), or equivalently, the residuals r_{F3} and r_{N2} can be used for distinguishing between these two modes.

[CE8] New counter-example:

- $b_f = N1 \to F1 \to F2 \to F3\ (\to F1 \to F2 \to F3)^\infty$,
- $b_N = N1 \to N1 \to N2 \to N3\ (\to N1 \to N2 \to N3)^\infty$.

But, $F3$ can be distinguished from $N3$ because the corresponding invariant sets do not intersect according to (2), or equivalently, the residuals r_{F3} and r_{N3} can be used for distinguishing between these two modes.

[CE9] No more counter-example.

By analysing this diagnosability proof, we notice that we need to distinguish the following pairs of modes: $(F3,N1)$, $(F3,N2)$, $(F3,N3)$, and $(F2,N1)$. However, we do not need any other distinguishability power. For instance, we do not need to distinguish $N1$ from $N2$. This may look obvious (and, in this example, not particularly useful), but notice that this result was derived automatically.

Note also that the discernibility pairs are sufficient for diagnosability; they are not always necessary. In the example, the pair $(F2,N1)$ is unnecessary.

3 Preliminaries on Diagnosability

Because diagnosability strongly hinges on what diagnosis algorithm is used and because our diagnosis algorithm uses a model of the system, we start by providing a definition of diagnosability based on models. Next we quickly discuss how abstract models can be used to test diagnosability.

The definitions presented here are meant to be generic and to apply to both hybrid and discrete event setups.

3.1 Model-Based Diagnosability

We call "model", hereafter denoted M, the implicit representation of a set of "system behaviours" (both faulty and nominal), where a system behaviour, denoted $\sigma \in M$, represents the evolution of the system state during a (finite or infinite) time window. The model is assumed to be prefix-closed (if a behaviour is possible, its prefixes are possible) and live (all behaviours have a future). We also assume a single fault, although the generalization to multiple fault is straightforward. We shall use $N = \emptyset$ as a shortcut for "nominal" and $F = \Sigma_\mathrm{f}$ as a shortcut for "faulty". We write $M[N]$ and $M[F]$ the subsets of nominal and faulty behaviours of M.

A model M is equipped with an observation function obs_M that indicates what can be observed when a specified behaviour takes place: $o \in obs_M(\sigma)$ is one of the possible system observations for the behaviour $\sigma \in M$. It is assumed that the observation function satisfies natural assumptions such as the fact that if σ' has a prefix σ, then every observation $o' \in obs_M(\sigma')$ has a prefix $o \in obs_M(\sigma)$. To simplify notations, we drop the references to the model and simply write $obs(\sigma)$ when not ambiguous.

Model-Based Diagnosis is the problem of deciding whether the observations generated by the system betray a nominal or a faulty behaviour. Specifically, given a model M, given an unknown behaviour $\hat{\sigma} \in M$, given the observation $\hat{o} \in obs(\hat{\sigma})$, the model-based diagnosis is defined as follows:

$$\Delta(\hat{o}) = \{\delta \in \{N,F\} \mid \exists \sigma \in M[\delta].\ \hat{o} \in obs(\sigma)\},$$

i.e., the hypotheses (N or F) for which there exists a behaviour consistent with the observations. Under the usual assumptions that the diagnosis model is complete (all possible system behaviours are in M and all possible observations of every behaviour σ are in $obs(\sigma)$) the diagnosis is guaranteed to be correct: the hypothesis $\hat{\delta}$ that actually affects the system appears in $\Delta(\hat{o})$.

Diagnosability is the property that if a fault occurs on the system, then this fault will eventually be diagnosed. Because we limit ourselves to a single fault, this implies that the diagnosis will eventually be $\Delta = \{F\}$.

We use the notation $\sigma \sqsubseteq \sigma'$ to specify that σ is a prefix of σ', and $\sigma \sqsubseteq_d \sigma'$ to specify that the time window of σ' is at least d units of time longer than that of σ.

Definition 1. *The model M is* diagnosable *if the following property holds:*

$$\forall \sigma \in M[F].\ \exists d \in \mathbf{N}.\ \forall \sigma' \in M.\ \sigma \sqsubseteq_d \sigma'$$

$$\Rightarrow \forall o' \in obs(\sigma').\ \Delta(o') = \{F\}.$$

In words, this definition states that for any faulty behaviour σ, after waiting for a sufficiently long time (d, leading to extended behaviour σ' and observation o'), the diagnosis is unambiguously F.

3.2 Abstraction and Diagnosability

Abstraction plays an important role in this work. The idea of abstraction is to remove some information included in the model in order to make the task of diagnosis, or diagnosability, computationally simpler, or even decidable.

A model M' is an *abstraction* of M if the former allows for more behaviours than the latter:

$$M' \supseteq M \ \wedge\ (\forall \sigma \in M.\ obs_{M'}(\sigma) \supseteq obs_M(\sigma)).$$

M is then called a *refinement* of M'. Notice that we already mentioned that the diagnosis model should be an abstraction of the system model.

Abstraction can help prove diagnosability through the following lemma.

Lemma 1. *Let M' be an abstraction of M. If M' is diagnosable, then M is also diagnosable.*

This lemma can be easily proved by noting that the condition for diagnosability (Definition 1) is easier to satisfy for the refined models.

Notice that Lemma 1 does not tell us much about diagnosability of M if M' is not diagnosable.

4 Hybrid systems and DES Abstraction

We first introduce the definition of hybrid systems that we are considering in this paper. We then move to the discrete event model. We review some results about verifying diagnosability of DES. Finally, we show how the hybrid system can be abstracted to a DES.

4.1 Hybrid Systems

A hybrid system is a system whose behavior interlinks discrete and continuous dynamics. Discrete dynamics are represented by a set of discrete states, also called "modes". Each mode has specific continuous dynamics, represented by a set of algebraic differential equations which constrain the continuous state, input and output variables. Input and output variables are measured. Transitions between modes trigger upon discrete events or guards (conditions on the continuous state). In this paper, we consider uncertain hybrid systems that can be represented by uncertain hybrid automata ([Lunze and Lamnabhi-Lagarrigue, 2009]) :

$$H = (Q, T, \zeta, C, (q_0, \zeta_0)) \tag{3}$$

where:

- Q is the set of discrete system states, i.e. modes, with $|Q| = m$. Each state $q \in Q$ represents a mode of operation of the system.
- $T \subseteq Q \times Q$ is the set of *transitions*. A transition $t(q_i, q_j)$ may be guarded by a condition given as a set of equations $\mathscr{G}(t(q_i, q_j)) = g_{ij}(x, \theta_g) = 0$, θ_g being a constant parameter vector. The transition happens when the state $x(t)$ hits the guard g_{ij}. A reset map $\mathscr{R}_{ij}$, possibly equal to the identity, is specified.
- ζ is the set of continuous variables, functions of time t, including state, input, and output variables as defined below. Input/output variables form the set of observable, i.e. measured, continuous variables denoted by ζ_{OBS}.
- $C = \{C_q\}$ is the set of system constraints linking continuous variables in mode q:

$$\begin{cases} \dot{x}(t) = f_q(x(t,p), u(t), p) \\ y(t) = g_q(x(t,p), p) \\ x(t_0) = x_0 \in X_0 \\ p \in P \subseteq \mathbb{R}^p \\ t_0 \leq t \leq T \end{cases} \tag{4}$$

  where :
  - $x(t) \in \mathbb{R}^{n_x}$ and $y(t) \in \mathbb{R}^{n_y}$ denote the vectors of state and output variables at time t respectively,
  - $u(t) \in \mathbb{R}^{n_u}$ is the vector of input variables at time t,
  - the functions f_q and g_q are real and analytic on $D_x \subseteq \mathbb{R}^{n_x}$, where D_x is the definition domain of $x(t)$ such that $x(t) \in D_x$ for every $t \in [t_0,\ T]$ and $p \in P$, T is a finite or infinite time bound,
  - the vector of parameters p is assumed to belong to a connected set $P \subseteq \mathbb{R}^p$,
  - the initial condition $x(t_0) = x_0$ is assumed to belong to a connected set $X_0 \subseteq \mathbb{R}^{n_x}$.
- (q_o, ζ_o) is the initial condition of the hybrid system, where ζ_o is the initial valuation of the continuous variables of ζ and $q_o \in Q$.

Transitions from one mode to another change the continuous dynamics driving the behavior of the system.

4.2 Discrete Event Systems

A discrete event system is a discrete state, event driven system where the state evolution depends on the occurrence of asynchronous discrete events. For consistency with hybrid system, these states are here referred to as "modes".

Compared to hybrid systems, DES have discrete observations. Contrary to the standard literature [Sampath *et al.*, 1995; Lamperti and Zanella, 2003; Pencolé and Cordier, 2005], we assume that the observations are state-based, but this choice is purely for convenience: there is no fundamental difference between state-based and event-based observations. We assume a constant set I of *indicators* which are observable properties about the current system mode—when the property holds in the mode, we say that the indicator is satisfied. In a given mode, an indicator could be always satisfied, never satisfied, or sometimes satisfied.

Definition 2. *A* discrete event system *is a tuple* $D = \langle Q, T, q_o, L, Eph \rangle$ *where* Q *is a set of* modes *with* $q_o \in Q$ *the* initial mode*,* $T \subseteq Q \times Q$ *is the set of* transitions*,* $L : M \times I \to \{0, 1, -1\}$ *is the* indicator function*, and* $Eph \subseteq 2^Q$ *is a collection of* ephemeral sets.

A behaviour on the discrete event system is a sequence of modes $q_0 \to \ldots \to q_k$ such that $q_0 = q_o$ and all $\langle q_{i-1}, q_i \rangle$ are transitions.

For every mode $q \in Q$ and every indicator $indi \in I$, $L(q, indi) = 1$ (resp. $L(q, indi) = -1$) specifies that the indicator is always (resp. never) satisfied in this mode. We define $I^{>0}(q) = \{indi \in I \mid L(q, indi) > 0\}$ as the list of indicators that are always satisfied in mode q and $I^{\geq 0}(q) = \{indi \in I \mid L(q, indi) \geq 0\}$ the list of indicators that are always or sometimes satisfied. Then an observation θ in mode q is the list of indicators satisfied in this mode and is such that:

$$I^{>0}(q) \subseteq \theta \subseteq I^{\geq 0}(q).$$

Notice that at different time, the observation of the same mode may vary (but it always satisfies the subset constraint above). An observation of $q_0 \to \ldots \to q_k$ is then a sequence $o = \theta_0, \ldots, \theta_k$ where each θ_i is an observation of q_i.

We now explain the last parameter of the DES definition. A DES is event driven, meaning that the mode of the system may remain the same over time. This is allowed via an explicit transition $\langle q, q \rangle \in T$ for all $q \in Q$. However, there are modes in which one cannot stay forever. For example, in a situation where a container is being filled at a non-trivial rate, the system mode will eventually change (as e.g. the container will become full, or it will start leaking). We model this with a notion of *ephemerality*. Formally for any infinite sequence $q_0 \to q_1 \to \ldots$ let us denote Q_∞ the set of modes that appear infinitely often; then this set of modes cannot appear in Eph:

$$Q_\infty \notin Eph.$$

This property is similar to that of *fairness* frequently used in model checking but also in diagnosis [Biswas *et al.*, 2010]. Notice that if a set $X \subseteq Q$ is ephemeral then any subset of X should be ephemeral too ($X \in Eph \wedge X' \subseteq X \Rightarrow X' \in Eph$).

4.3 Diagnosability of DES

Diagnosability of DES is a well-studied problem. It was introduced by Sampath et al. [Sampath *et al.*, 1995] and polynomial algorithms were developed in parallel by Yoo and Lafortune [Yoo and Lafortune, 2002] and Jiang et al. [Jiang *et al.*, 2001]. These papers assume event-based observations, but state-based observations can be considered too. Similarly it is possible to include fairness conditions [Grastien, 2009].

The standard way to solve problems of diagnosability of DES is to search two infinite behaviours in the model, the first of which is faulty and the second is not, and that are observation-similar. This search is implemented by constructing the twin plant (defined below) and searching for reachable fair cycles, i.e., cycles that do not remain in ephemeral sets of modes. We assume that the modes are partitioned into nominal modes Q_N and faulty modes Q_F (when faults are defined as events, we assume that every mode remembers whether a faulty event occurred).

Definition 3. *Given the DES* $D = \langle Q, T, q_o, L, Eph \rangle$ *the* twin plant *is the state machine* $\langle \mathfrak{Q}, \mathfrak{T}, \mathfrak{q}_o, \mathfrak{E} \rangle$ *where:*

- $\mathfrak{Q} = \{\langle q_1, q_2 \rangle \in Q \times Q \mid \forall indi \in I.\ \{L(q_1, indi), L(q_2, indi)\} \neq \{-1, 1\}\}$*,*
- $\mathfrak{T} = \{\langle \langle q_1, q_2 \rangle, \langle q'_1, q'_2 \rangle \rangle \in \mathfrak{Q} \times \mathfrak{Q} \mid \langle q_1, q'_1 \rangle \in T \wedge \langle q_2, q'_2 \rangle \in T\}$*,*
- $\mathfrak{q}_o = \langle q_o, q_o \rangle$*, and*
- $\mathfrak{E} = \{\mathfrak{Q}' \subseteq \mathfrak{Q} \mid \exists X \in Eph \text{ where } X = \{q \mid \langle q, q' \rangle \in \mathfrak{Q}'\}\}$*.*

The first item in Definition 3 simply indicates that the twin plant only includes states $\mathfrak{q} = \langle q_1, q_2 \rangle$ such that the two modes q_1 and q_2 do not disagree on any indicator. Indeed if $\{L(q_1, indi), L(q_2, indi)\} \neq \{-1, 1\}$ then the indicator is always satisfied in one mode and always unsatisfied in the other mode.

Notice that the indicators do not appear in the twin plant as they are only relevant to define its states $\mathfrak{Q}$. Notice also that the ephemerality relation on the twin plant is defined only on the first element of the twin plant: a set of states $\mathfrak{Q}'$ of the twin plant is ephemeral iff the set X of modes that are mentioned in the states of $\mathfrak{Q}'$ (as first element of the pair) is ephemeral. Accordingly given a cycle $\mathfrak{q}_1 \to \ldots \to \mathfrak{q}_i$ (i.e., such that $\mathfrak{q}_1 = \mathfrak{q}_i$), we say that this cycle is *fair* iff $\{\mathfrak{q}_1, \ldots, \mathfrak{q}_i\} \notin \mathfrak{E}$.

Proposition 1. *[Jiang* et al.*, 2001; Yoo and Lafortune, 2002] Let* $\mathfrak{A} = (Q_F \times Q_N) \cap \mathfrak{Q}$ *be the set of* ambiguous states *of the twin plant. The DES is diagnosable iff the twin plant does not contain any fair cycle of ambiguous states that can be reached from its initial state.*

The cycle $\mathfrak{c} = \mathfrak{q}_1 \to \ldots \mathfrak{q}_j$ mentioned in Proposition 1 (where $\mathfrak{q}_i = \langle q_i, q'_i \rangle$ for each index i) is called the *counter-example*. It represents a possible faulty system behaviour (namely $q_1 \to q_2 \to \dots$) that can be mistaken for a nominal behaviour (namely $q'_1 \to q'_2 \to \dots$). We write $\mathfrak{Q}^\infty(c)$ the list of (twin plant) states that appear in the cycle.

5 Diagnosability of Hybrid Systems with DES Methods

In this section we reduce the problem of diagnosability of hybrid systems to the problem of diagnosability of DES. To this end we define D^∞_M, a DES abstraction of the hybrid system. All the continuous dynamics of the hybrid system are embedded in the indicators I (c.f. Section 5.1) and ephemerality properties (c.f. Section 5.2) of the DES. Then from Lemma 1 diagnosability of D^∞_M implies diagnosability of the hybrid system. Again, nondiagnosability of D^∞_M leaves open the question of diagnosability of the hybrid system.

5.1 Residuals, discernibility and the indicator function

Discernibility of a pair of modes can be verified from the *residuals* attached to the modes [Bayoudh and Travé-Massuyès, 2014] [Vento *et al.*, 2015]. Residuals are consistency indicators used by the FDI community that allow checking the observations against the continuous dynamics of every mode.

Residual generators can be obtained from *analytical redundancy relations*, or ARR for short, obtained themselves from the continuous differential models (4) associated to every mode. There are several approaches for generating ARRs [Blanke and Kinnaert, 2016] (as e.g., using structural methods, or decoupling unknown variables), all of them based on the elimination of the unknown variables $x(t)$. The elimination process produces a testable expression that only depends on variables that can be determined from measured variables, i.e. input and output variables $u(t)$ and $y(t)$ and their derivatives $\bar{y}^{(n)}(t)$ and $\bar{y}^{(n)}(t)$ up to the order n. Thus, in the ideal case (no noise and uncertainty)[1], as long as the hybrid system is actually in mode q and there is no fault, the residual $r(\bar{y}^{(n)}(t), \bar{u}^{(n)}(t))$ (in vector form) satisfies

$$r(\bar{y}^{(n)}(t), \bar{u}^{(n)}(t)) = 0 \tag{5}$$

Otherwise, the residual $r(\bar{y}^{(n)}(t), \bar{u}^{(n)}(t))$ (or r for short) is different from zero indicating that observations are inconsistent with the continuous dynamics of mode q.

In this paper, discernibility is not explicit. It is represented in the first item of the Definition 3 of the twin plant. But, the set of indicators I is built from the residuals generated for every mode. An indicator $indi \in I$ is associated to the residual vectors obtained for every mode. Denote by r_q the residual vector for mode q. Then, from the properties of residuals, we have :

- $L(q, indi) = 1$ because r_q evaluates to zero when the system is in mode q (unknown faults are not considered in this paper),
- $L(q', indi) = -1$ when q' is discernible from q because r_q never evaluates to zero when the system is in mode q',
- $L(q', indi) = 1$ when q' is not discernible from q because r_q then evaluates to zero when the system is either in mode q or in mode q',
- $L(q', indi) = 0$ when q' may be discernible from q or not (this may happen when the models are uncertain, cf. the example of Section 2).

5.2 Checking ephemerality

Ephemerality is a notion that, as far as we are aware, has never been introduced before. Ephemerality means that the system mode will always leave a given set of modes; comparatively the well-known *stability* asserts that the system will always stay within that set.

We believe that ephemerality is a problem that will require further investigation, but we propose two methods to check it:

- by running a hybrid *reachability* procedure from all possible initial states. This method is applicable to one mode as well as a set of modes,
- by computing the *positive invariant set*. This set represents the attractor state region, where the continuous dynamics drive the state and where the state stays forever. As far as we know, this method is applicable to one mode only.

The advantage of the invariant set theory is that it provides the theoretical attractor set for a dynamic model. However, it only gives static information, i.e. we know that the system converges towards this set. On the contrary, reachability analysis provides the atteignable set during transient behavior but it requires to make a sufficiently long run (whose minimal temporal bound is unknown) to obtain the invariant set.

Ephemerality via reachability

The reachability problem for hybrid automata is the problem of determining, given an automaton H, whether there is a trajectory of the transition system, that visits a hybrid state of the form (q, x^*).

In this paper, we are concerned by – possibly nonlinear – uncertain hybrid systems as given by (3). For these systems, continuous dynamics, guard sets and reset functions are defined by nonlinear functions and all uncertainties are considered bounded. These systems may be non-deterministic because the uncertainties that determine the initial, flow and jump conditions might not determine unique values for the state variables.

At some instant t, the hybrid state $(Q_t, x(t))$ is uncertain, which means that Q_t and $x(t)$ are set-valued, i.e. $Q_t \subseteq Q$ and $x(t) = x_t \subseteq \mathbb{R}^{n_x}$. Q_t is the set of "active" modes of the uncertain hybrid system at time t, i.e. the modes in which the system may operate at t,

[1] In case of noise or uncertainty the residual consistency is checked with statistical or set-membership methods [Blanke and Kinnaert, 2016]

and x_t is the set of possible states at t given an uncertain initial hybrid state at time t_0. In a given mode, the continuous state trajectory takes the form of a "flow-pipe" which defines the bounds of the continuous state in time. When analyzing hybrid systems, intersections with guard sets that enable discrete transitions may occur, and when a flow-pipe of non-zero size reaches a guard condition, there is a non-empty set of instants during which the constraints are satisfied, leading to a continuum of switching times.

Running a hybrid reachability procedure for a set of modes $X \subseteq Q$ from a given initial hybrid state provides a way to envision all the hybrid states reachable by the hybrid system, which can be used to assess that the system cannot stay forever in this set of modes. Indeed, we can assess that all the trajectories necessarily intersect one of the mode's guards, proving ephemerality for the set of modes [2].

Several methods have been developed recently for the explicit computation of reachable sets (see for instance [Asarin *et al.*, 1995; Girard, 2005; Kurzhanskiy *et al.*, 2007; Althoff *et al.*, 2008; Ramdani *et al.*, 2009]) but only a few of them consider hybrid systems and even less uncertain hybrid systems [Henzinger *et al.*, 2000; Ramdani and Nedialkov, 2011; Maïga *et al.*, 2015]. In this paper, we have used the method and associated software [3] presented in [Maïga *et al.*, 2015] that can be decomposed in three algorithmic steps:

- computing the reachable set when the system is in a given operation mode, which relies on a validated set integration method based on Interval Taylor Methods [Nedialkov *et al.*, 1999],
- computing the discrete transitions, i.e. detecting and localizing when (and where) the continuous flow-pipe intersects the guard sets, which is solved using interval constraint-propagation techniques,
- enclosing the multiple trajectories that result from an uncertain transition once the whole flow-pipe has transitioned, which is approached with zonotope bounding method.

Ephemerality via set-invariance

Another way of checking ephemerality is based on the use of set-invariance based on the positive invariant set concept [Seron *et al.*, 2012][Blanchini, 1999]. The computation of these sets can be performed beforehand and depends on known system's dynamics and bounds on input signals and disturbances/uncertainties.

Definition 4. *The set $\mathscr{S} \subset \mathbb{R}^n$ is said to be* positively invariant *w.r.t. the continuous dynamics of a mode q of the hybrid system if every solution trajectory $x(t)$ with initial condition $x(0) \in \mathscr{S}$ is globally defined and such that $x(t) \in \mathscr{S}$ for $t > 0$.*

In case of considering uncertainty in the parameters p of the continuous dynamics of the mode q, the previous concept can be straightforwardly extended to consider the uncertainty effect by defining the *robust positive invariant (RPI) set* [Blanchini, 1999]. The *minimal robust positive invariant* (mRPI) set is defined as the RPI set contained in any closed RPI set.

In general, exact invariant sets are very difficult to compute. However, they can be approximated by using simple sets (ellipsoids, zonotopes, polytopes) and associated set-arithmetic operations using an iterative procedure [Blanchini, 1999].

The application of the set-invariance approach to prove that one mode q is ephemeral comes back to check whether all the states in the mRPI set of a mode satisfy the guard [4]. Using set-invariance theory to prove ephemerality of a set of modes is not as convenient and reachability analysis is preferred in this case.

6 Incremental Diagnosability

We now present how to test diagnosability incrementally, i.e., by starting with abstract L and Eph parameters and incrementally refining them until diagnosability has been shown, or an irrefutable counter-example was produced. There are two main benefits for using an incremental approach:

1. Computing precisely D_M^∞ can be very expensive (it requires computing Eph for instance).
2. Computing L incrementally allows us to calculate a subset of the indicators sufficient to guarantee diagnosability (as well as what sensors to monitor), akin to optimal observability [Brandán Briones *et al.*, 2008].

6.1 Description of the Approach

Our approach is summarised in Algorithm 1. We start with a hybrid system model M. From this model we generate the most abstract DES model D_M^0 (described in Section 6.2). We check diagnosability of the current abstract model. If it is diagnosable, then we found an abstraction of M that is diagnosable. In other words, we found an abstraction that allows us to diagnose precisely the system. If the current model is not diagnosable, then we analyse the counter example generated by the diagnosability procedure and search for a refinement of L or Eph that contradicts the counter example. If no refinement is found, then we cannot prove that the system is diagnosable. If a refinement is found, then it is included to the abstract model and we test diagnosability of this new model again.

[2] Let us notice that reachability analysis can also be used to check discernibility. Indeed, if reachability analysis is run for two modes starting with all their possible initial states, if the flow-pipes separate at some point in time, it means that these modes are discernible.

[3] available for download at `http://projects.laas.fr/ANR-MAGIC-SPS/`

[4] Let us notice that set-invariance can also be used to check discernibility. Indeed, two modes that have mRPI sets that do not intersect are discernible.

Algorithm 1 Diagnosability

```
Input: hybrid system model $M$
$A := D^0_M$
loop
    if $A$ is diagnosable then        // Using the Twin Plant method and Lemma 1
        return diagnosable (with abstraction $A$)
    end if
    let $c$ be a counter example for $A$
    if there is a refinement of $A$ that contradicts $c$ then
        apply refinement to $A$
    else
        return could not prove diagnosability
    end if
end loop
```

6.2 Abstraction through L and *Eph*

The correctness of our approach relies on the following two properties.

Lemma 2. *Let $D = \langle Q, T, q_o, L, Eph\rangle$ be a DES model and let L' be a function satisfying $L'(q, indi) \neq 0 \Rightarrow L(q, indi) = L'(q, indi)$ for all pair modes–indicator. Then, the model $D' = \langle Q, T, q_o, L', Eph\rangle$ is an abstraction of D.*

Lemma 3. *Let $D = \langle Q, T, q_o, L, Eph\rangle$ be a DES model and let Eph' be a subset of Eph. Then, the model $D' = \langle Q, T, q_o, L, Eph'\rangle$ is an abstraction of D.*

Both lemmas are trivial: L' allows for more observations while Eph' allows for more (infinite) behaviours.

We now define D^0_M, the abstract DES that we begin our algorithm with. This model remembers the list of modes and how to transition from one mode to the next, but it forgets the information about what is observed in every state and what the ephemeral sets are. Formally, $D^0_M := \langle Q^0, T^0, q^0_o, L^0, Eph^0\rangle$ is defined from $D^\infty_M = \langle Q, T, q_o, L, Eph\rangle$, where

- $Q^0 = Q$,
- $T^0 = T \cup \{\langle q, q\rangle \mid q \in Q\}$,
- $q^0_o = q_o$,
- $L^0(q, indi) = 0$ for all pair $\langle q, indi\rangle$,
- and $Eph^0 = \emptyset$.

Clearly, D^0_M is an abstraction of D^∞_M.

6.3 Contradicting a Counter Example

Consider two DES D^1 and D^2 such that D^1 is an abstraction of D^2 and D^2 is diagnosable while D^1 is not. A cycling counter example $\mathfrak{c} = \mathfrak{q}_0 \to \ldots \to \mathfrak{q}_j$ (where $\mathfrak{q}_k = \langle q_k, q'_k\rangle$ for all index k) exists on the twin plant of D^1 that does not exist on the twin plant of D^2.

It is easy to see that there are two possible reasons why such a counter example could exist:

1. Either there exists an index $\ell \in \{0, \ldots, j\}$ such that $\mathfrak{q}_\ell \in \mathfrak{Q}^1 \setminus \mathfrak{Q}^2$. This implies $\{L^2(q_\ell, indi), L^2(q'_\ell, indi)\} = \{-1, 1\} \neq \{L^1(q_\ell, indi), L^1(q'_\ell, indi)\}$ for some indicator $indi \in I$.
2. Or $\{\mathfrak{q}_i, \ldots, \mathfrak{q}_j\} \in \mathfrak{E}^2 \setminus \mathfrak{E}^1$. This implies that $\{q_i, \ldots, q_j\}$ (the first modes of all state $\mathfrak{q}_i, \ldots, \mathfrak{q}_j$) belongs to Eph^2 and not to Eph^1.

Given the counter example, we search for a reason to reject it. If we find one, we refine the model (by updating the function L or the set *Eph*) so that this counter example will not be generated at the next iteration. If we cannot find one, then we cannot prove diagnosability, although whether this is: i) a fundamental problem of the hybrid model or ii) a consequence of the abstraction to a discrete event setting is unknown.

It is then easy to demonstrate the following theorem.

Theorem 1. *If D^∞_M is diagnosable and the search for refinements contradicting counter examples is complete, then Algorithm 1 always returns an abstraction of D^∞_M that is diagnosable.*

7 Conclusion

This paper has addressed the problem of determining the diagnosability of hybrid systems by abstracting hybrid models to a discrete-event setting. The abstracted model only remembers two pieces of information: indistinguishability between modes (when they are guaranteed to generate different observations) and ephemerality (when the system cannot stay forever in a given set of modes). Then, standard DES diagnosability algorithms are applied to the abstracted model. An iterative approach to diagnosability is proposed that starts with the most abstract DES model and iteratively calls for refinements until diagnosability is proved or there are no more refinements available. The proposed approach has been illustrated with an academic example that clearly shows how the different techniques used interplay.

Acknowledgments

The authors want to thank you Moussa Maïga for helping in running the hybrid reachability scenarios.

References

[Althoff *et al.*, 2008] Matthias Althoff, Olaf Stursberg, and Martin Buss. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 4042–4048. IEEE, 2008.

[Asarin *et al.*, 1995] Eugene Asarin, Oded Maler, and Amir Pnueli. Reachability analysis of dynamical systems having piecewise-constant derivatives. *Theoretical computer science*, 138(1):35–65, 1995.

[Basseville *et al.*, 2001] M. Basseville, M. Kinnaert, and Nyberg M. On fault detectability and isolability. *European Journal of Control)*, 7(6):625–641, 2001.

[Bayoudh and Travé-Massuyès, 2014] M. Bayoudh and L. Travé-Massuyès. Diagnosability analysis of hybrid systems cast in a discrete-event framework. *Discrete Event Dynamic Systems*, 24(3):309–338, 2014.

[Biswas *et al.*, 2010] S. Biswas, D. Sarkar, S. Mukhopadhyay, and A. Patra. Fairness of transitions in diagnosability of discrete event systems. *Journal of Discrete Event Dynamical Systems (JDEDS)*, 20:349–376, 2010.

[Blanchini, 1999] Franco Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999.

[Blanke and Kinnaert, 2016] M. Blanke and M. Kinnaert. *Diagnosis and Fault-tolerant control*. Springer, 2016.

[Brandán Briones *et al.*, 2008] L. Brandán Briones, A. Lazovik, and Ph. Dague. Optimal observability for diagnosability. In *Nineteenth International Workshop on Principles of Diagnosis (DX-08)*, pages 31–38, 2008.

[Cocquempot *et al.*, 2004] V. Cocquempot, T.E. Mezyani, and M. Staroswiecki M. Fault detection and isolation for hybrid systems using structured parity residuals. In *Asian Control Conference*, pages 1204–1212, 2004.

[Daigle *et al.*, 2008] M. Daigle, X. Koutsoukos, and G. Biswas. An event-based approach to hybrid systems diagnosability. In *Nineteenth International Workshop on Principles of Diagnosis (DX-08)*, pages 47–54, 2008.

[Girard, 2005] Antoine Girard. Reachability of uncertain linear systems using zonotopes. In *Hybrid Systems: Computation and Control*, pages 291–305. Springer, 2005.

[Grastien, 2009] A. Grastien. Symbolic testing of diagnosability. In *20th International Workshop on Principles of Diagnosis (DX-09)*, pages 131–138, 2009.

[Henzinger *et al.*, 2000] Thomas A Henzinger, Benjamin Horowitz, Rupak Majumdar, and Howard Wong-Toi. Beyond hytech: Hybrid systems analysis using interval numerical methods. In *Hybrid systems: Computation and control*, pages 130–144. Springer, 2000.

[Jiang *et al.*, 2001] S. Jiang, Z. Huang, V. Chandra, and R. Kumar. A polynomial algorithm for diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control (TAC)*, 46(8):1318–1321, 2001.

[Kurzhanskiy *et al.*, 2007] Alex A Kurzhanskiy, Pravin Varaiya, et al. Ellipsoidal techniques for reachability analysis of discrete-time linear systems. *IEEE Transactions on Automatic Control*, 52(1):26–38, 2007.

[Lamperti and Zanella, 2003] G. Lamperti and M. Zanella. *Diagnosis of active systems*. Kluwer Academic Publishers, 2003.

[Lunze and Lamnabhi-Lagarrigue, 2009] J. Lunze and F. Lamnabhi-Lagarrigue, editors. *Handbook of Hybrid Systems Control : Theory, Tools, Applications*. Cambridge University Press, Cambridge, UK, New York, 2009.

[Maïga *et al.*, 2015] Moussa Maïga, Nacim Ramdani, Louise Travé-Massuyès, and Christophe Combastel. A comprehensive method for reachability analysis of uncertain nonlinear hybrid systems. *Automatic Control, IEEE Transactions on, ISSN 0018-9286, doi:10.1109/TAC.2015.2491740*, PP(99), 2015.

[Nedialkov *et al.*, 1999] Nedialko S Nedialkov, Kenneth R Jackson, and George F Corliss. Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105(1):21–68, 1999.

[Pencolé and Cordier, 2005] Y. Pencolé and M.-O. Cordier. A formal framework for the decentralised diagnosis of large scale discrete event systems and its application to telecommunication networks. *Artificial Intelligence (AIJ)*, 164(1–2):121–170, 2005.

[Ramdani and Nedialkov, 2011] Nacim Ramdani and Nedialko S Nedialkov. Computing reachable sets for uncertain nonlinear hybrid systems using interval constraint-propagation techniques. *Nonlinear Analysis: Hybrid Systems*, 5(2):149–162, 2011.

[Ramdani *et al.*, 2009] Nacim Ramdani, Nacim Meslem, and Yves Candau. A hybrid bounding method for computing an over-approximation for the reachable set of uncertain nonlinear systems. *Automatic Control, IEEE Transactions on*, 54(10):2352–2364, 2009.

[Sampath *et al.*, 1995] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis. Diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control (TAC)*, 40(9):1555–1575, 1995.

[Seron *et al.*, 2012] M. M. Seron, J. A. De Dona, and S. Olaru. Fault tolerant control allowing sensor healthy-to-faulty and faulty-to-healthy transitions. *IEEE Transactions on Automatic Control*, 57(7):1657–1669, July 2012.

[Vento *et al.*, 2015] J. Vento, L. Travé-Massuyès, V. Puig, and R. Sarrate. An incremental hybrid system diagnoser automaton enhanced by discrenibility properties. *IEEE Transactions on Systems, Man, and Cybernetics (TSMC)*, 45(5):788–804, 2015.

[Yoo and Lafortune, 2002] T.-S. Yoo and S. Lafortune. Polynomial-time verification of diagnosability of partially observed discrete-event systems. *IEEE Transactions on Automatic Control (TAC)*, 47(9):1491–1495, 2002.