This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

Verification of Hybrid Automata Diagnosability with Measurement Uncertainty

Yi Deng, Alessandro D'Innocenzo, Maria D. Di Benedetto,

Stefano Di Gennaro, and A. Agung Julius

***Abstract*—The problem of system diagnosability verification is concerned with whether a fault in the system operation can be diagnosed by using the system model and observations of the system output. In this paper, we investigate the (δ_d, δ_m)-diagnosability of hybrid automata, which characterizes the maximum delay for diagnosing faults since their first occurrence, given the measurement uncertainty in observing the system output.**

We present a methodology that analyzes the (δ_d, δ_m)-diagnosability of hybrid automata. Due to the complex dynamics, the hybrid system diagnosability is often difficult to analyze directly. We thus propose an approach of constructing an abstraction using the trajectories of the original system. Their (δ_d, δ_m)-diagnosability properties are proved to be quantitatively related to each other. The abstraction has only finitely many trajectories that extend to the end of the time horizon of interest, so its diagnosability can be easily calculated, and then used to derive the diagnosability of the original system. We illustrate this procedure with an example.

I. Introduction

For safety and economic considerations, diagnosing faulty states of system operation plays an import role in industry. For an aircraft, for example, we want the system to be able to locate an abrupt engine malfunction in time to take remedial action so that a catastrophic accident is avoided; or for a household air purifier, we want its filter screen degradation to be tracked and indicated to the user for a replacement; etc. This paper presents fault diagnosis for systems that are modeled as hybrid automata, by only observing ordered discrete events with their timing, which is assumed to be inaccurate to some extent. We reduce the problem to a finite one, and practically compute the diagnosability as well as the diagnoser.

Fault diagnosis involves three tasks: detection (whether something goes wrong), isolation (where it goes wrong), identification (what size the fault has), while the former two deserve primary emphasis [1]. Thus the essential task consists in discriminating a fault from normal system behaviors as well as from other faults. Based on the system model, this can be fulfilled by comparing available measurements with information analytically derived from the mathematical model of systems. As modeled, a fault has some particular pattern of anticipated measurements as its symptoms, which generate the *fault signature* [2]. Intuitively, fault diagnosability is determined by the discriminability of fault signatures.

Efficient fault diagnosis requires fault signatures that make good use of the measurement and computation resources to increase the signature discriminability. Indeed, when a system operates, only a limited source of information is available for measurement. Such considerations for hybrid systems motivates our research work in this paper. By definition, a hybrid system has interacting discrete and continuous dynamics. The system state can flow continuously, and can also jump by triggering an event. We want to diagnose faults for hybrid systems based on the discrete event sequences for the following reasons: First, without tracking discrete events, the system model and measurement information may be underutilized. Second, for a hybrid system where multiple modes (discrete states, also called locations) exist, if one does not track events, a mechanism that works in one location to diagnose faults may be invalid when the location changes. Third, it is presented in this paper that observing the timed event sequences can be sufficient for diagnosing faults. Consider a bouncing ball for instance. Suppose the only sensor is human ears that hear the sound when the ball hits the ground. By just tracking a sequence of sounds, one can tell whether the ball leaks, or is subjected to a sudden impact, etc. The time intervals between the sounds here are informative for fault diagnosis.

To study the fault diagnosability analysis problem for hybrid systems, we use some ideas on fault detection and isolation from the FDI (fault detection and isolation) community. There, information from measurement and model is synthesized in so-called residuals [3], which serve as the basis for diagnosis. A fault corresponds to a particular pattern of residuals (fault signature). So the diagnostic problem comprises of residual generation and decision making stages [4], see Fig.1 from [5]. This approach has wide application in continuous-time and discretized dynamical system models [6], [7]; but for hybrid systems, diagnosing faults in this way does not work efficiently due to the difficulty in capturing event sequences.

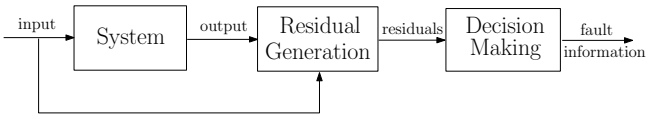


Fig. 1. Fault diagnosis in FDI community [5].

On the other hand, it is presented in [8], [9] that a hybrid system can be cast into the discrete event system (DES) framework so that approaches for DES diagnosis based on the order of events [10] can be drawn on. The present paper also reports on the hybrid system diagnosability problem relying on ordered events, but besides the event sequences we also want to make use of another property of the traces of the system that is closely related to the continuous dynamics, i.e., the time intervals between observed events. With timed behaviors considered, better diagnosability can be achieved due to the

Y. Deng and A. D'Innocenzo contribute equally to this paper.

YD and AAJ would like to acknowledge the support of NSF CAREER grant CNS-0953976.

The research leading to these results has received funding from the Italian Government under Cipe resolution n.135 (Dec. 21, 2012), project *INnovating City Planning through Information and Communication Technologies* (INCIPICT).

Y. Deng and A. A. Julius are with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute. E-mail={dengy3@rpi.edu; agung@ecse.rpi.edu}. Pin={99071, 37644}.

A. D'Innocenzo, M. D. Di Benedetto, and S. Di Gennaro are with the Department of Engineering and Information Sciences and Mathematics, Center of Excellence DEWS, University of L'Aquila, Italy. E-mail={alessandro.dinnocenzo@univaq.it; mariadomenica.dibenedetto@univaq.it; stefano.digennaro@univaq.it}. Pin = { 98620, 1545, 9603}.

Correspondence address: Yi Deng, Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, 110 Eighth Street, Troy, NY USA 12180.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

discriminability increased by the temporal properties; relevant works can be found in [11], [12].

Due to the limited measurement and computation resource, not every event is observable, and moreover the *measurement uncertainty* can be introduced thereby. It is clear that the inaccuracy of actual measurements can impair the discriminability of observed symptoms. Fault diagnosability is then affected. We thus formally formulate the fault diagnosability analysis problem in this paper for hybrid automata as follows: Suppose for the system, one can only observe events with their timing. Moreover, only a subset of the events are modeled as observable, and the timing has limited measurement accuracy. Any event that causes the system state to enter a faulty set is called a faulty event. We want to analyze whether the occurrence of any faulty event can be deduced within a limited time.

With this setting, sequences of timed observable events serve as the information available for fault diagnosis. In other words, symptoms are given by them; but due to the intrinsic complexity of hybrid dynamics, it can be rather difficult to analyze their discriminability directly. As illustrated in later sections, such analysis requires a pairwise comparison of all the possible observed event sequences, including comparing the events themselves and their timing. In general this is impossible for hybrid systems. We thus present a methodology that solves the problem in an indirect way: In Section II, we propose to construct for the hybrid automaton a system abstraction, whose diagnosability is very easy to analyze; in Section III, a quantitative relation between the diagnosability of the trajectories of the original system and the abstraction is proved, which allows us to derive the former from the later. An example is presented to illustrate the methodology.

The present paper uses (δ_d, δ_m)-diagnosability definition for hybrid automata that characterizes the limitation on the time delay for a fault to be diagnosed, as well as the uncertainty in measuring time intervals (see Section III-B). The (δ_d, δ_m)-diagnosability definition is initially proposed in [13], which we extend from faulty detectability to both detectability and isolability. The quantitative relation between the diagnosability of a hybrid automaton and its system abstraction can be found in Theorem 1. Another significant contribution of this paper is that we demonstrate a method to reduce a hybrid automaton to an abstraction of finitely many trajectories that extend to the end of the time horizon of interest (see Section II-C). This, in turn, allows us to practically diagnose faults that appear in the time horizon of interest. System abstraction has been studied previously for reachability analysis [14], [15], [16], [17], [18] and controller synthesis [19], [20], [21], [22], [23]. Compared to these literature, our work is different in the sense that: (i) We construct the abstraction for finite-horizon fault diagnosis; (ii) Our abstraction of hybrid automata does not grid the state space, so the computational cost does not necessarily grow exponentially as the state space dimension increases. The abstraction consists of only finitely many simulated trajectories. Its diagnosability can be easily calculated, and then used to analyze the diagnosability of the original hybrid automaton and construct the diagnoser.

II. Hybrid Automata Abstraction

A. Definition of Hybrid Automata

Hybrid automata are autonomous systems with interacting continuous and discrete dynamics, where the interaction between the continuous and discrete part can be described by events and invariant sets. We use the definition proposed by [24]:

Definition 1 (Hybrid Automaton)**.** *A hybrid automaton $H = (L \times X, L^0 \times X^0, D, E, Inv)$ is a tuple that consists of:*

- *A possibly infinite set $L \times X$ of hybrid states (ℓ, x), where $\ell \in L$ is the discrete state, and $x \in X$ is the continuous state. Discrete states are also called locations.*
- *A possibly infinite set $L^0 \times X^0$ of initial states.*
- *D associates with each location $\ell \in L$ the autonomous continuous time-invariant dynamics, $D_\ell : \dot{x} = D_\ell(x)$. We assume that this differential equation admits a unique global solution $\xi_\ell(t, x_\ell^0)$, where $\xi_\ell(0, x_\ell^0) = x_\ell^0$ is the initial condition in ℓ, and the function ξ_ℓ satisfies $\frac{\partial \xi_\ell(t, x_\ell^0)}{\partial t} = D_\ell(\xi_\ell(t, x_\ell^0))$.*
- *$Inv : L \to X$ associates with each location an invariant set $Inv(\ell) \subset X$. Only if the continuous state satisfies $x \in Inv(\ell)$, can the discrete state be at the location ℓ.*
- *E is a set of events. In each location ℓ, the system state evolves continuously according to D_ℓ until an event $e := (\ell, \ell', g, r), e \in E$ occurs. The event is guarded by $g \subset Inv(\ell)$. Namely, a necessary condition for the occurrence of e is $x \in g$. After the event, the discrete state changes from the source location ℓ to the target location ℓ', and the continuous state is reset according to the reset map $r : Inv(\ell) \to Inv(\ell')$. Let (ℓ, x) denote the system state that triggers $e = (\ell, \ell', g, r)$. Then the reset state is $(\ell', r(x))$.*

Let G_ℓ denotes the set union of guards such that the associated events all have ℓ as the source location. Let $\partial Inv(\ell)_{out}$ denote part of the boundary $\partial Inv(\ell)$ where the continuous state is evolving outward $Inv(\ell)$, i.e., given $\xi_\ell(\tau, x_\ell^0) \in \partial Inv(\ell)_{out}$, for any $t > 0$, there exists $t_1 \in (0, t)$ such that $\xi_\ell(\tau + t_1, x_\ell^0) \notin Inv(\ell)$. We adopt the following assumptions:

1) Non-deadlocking. For every location ℓ, we require $\partial Inv(\ell)_{out} \subset G_\ell$ to avoid deadlocking. Namely, whenever the continuous state is evolving outside $Inv(\ell)$, a jump must be specified.
2) Non-determinism. When the continuous state reaches a guard, an event may or may not occur (unless it is at $\partial Inv(\ell)_{out}$, where an event is forced to occur). Moreover, guards associated with different events can overlap, where any of the events may occur.
3) Well-posedness. The differential equation $\dot{x} = D_\ell(x)$ admits a unique solution, namely, it satisfies the Lipschitz condition.
4) The system does not have Zeno behavior [25].
5) All the reset maps are continuous functions.
6) The initial set is compact.

When the hybrid system operates, a sequence of events can be triggered. Some of the events are observable. while the rest are not. We associate each observable event with an observable output symbol $\psi \in \Psi_o$, and the unobservable events with the empty output symbol $\psi = \emptyset$. The initial state is not a reset state of an event $e \in E$, but it is assumed that we know when the system starts to operate, i.e., when to start the timer in the fault diagnosis. For convenience, we define an initialization event $e^0 \notin E$ associated with the output symbol ι (starting signal). Then a trajectory of the hybrid system can be defined as a sequence:

Definition 2 (Trajectory)**.** *Given $H = (L \times X, L^0 \times X^0, D, E, Inv)$, a trajectory of H is*

$$\rho = (e^0, \ell^0, x^0, \tau^0), (e^1, \ell^1, x^1, \tau^1) \cdots = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^N,$$

such that

- *for all $i \geq 0$, $(\ell^i, x^i) \in L \times X$, and $(\ell^0, x^0) \in L^0 \times X^0$;*
- *for all $i \geq 0$, $\tau^i \in \mathbb{R}_{\geq 0}$, and $\xi_{\ell^i}(t, x^i) \in Inv(\ell^i)$ for all $t \in [0, \tau^i]$;*
- *for all $i \geq 1$, $e^i = (\ell^{i-1}, \ell^i, g^i, r^i) \in E$, $\xi_{\ell^{i-1}}(\tau^{i-1}, x^{i-1}) \in g^i$, and $x^i = r^i(\xi_{\ell^{i-1}}(\tau^{i-1}, x^{i-1}))$, i.e., (ℓ^i, x^i) is the reset*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

state.

Suppose there is a trajectory $\rho' = \{(e'^i, \ell'^i, x'^i, \tau'^i)\}_{i=0}^{N'}$ *such that* $N' \leq N$, $(e'^i, \ell'^i, x'^i, \tau'^i) = (e^i, \ell^i, x^i, \tau^i)$ *for all* $i \in [0, N'-1]$, *and* $(e'^{N'}, \ell'^{N'}, x'^{N'}) = (e^{N'}, \ell^{N'}, x^{N'})$, $\tau'^{N'} \leq \tau^{N'}$, *then we call* ρ' *a sub-trajectory of* ρ.

Definition 3 (Timed Event Sequence)**.** *Given a trajectory* $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^{N}$, *the timed event sequence produced by* ρ *is*

$$p = (e^0, \tau^0), (e^1, \tau^1) \cdots = \{(e^i, \tau^i)\}_{i=0}^{N}.$$

Definition 4 (Label Sequence)**.** *Given a trajectory* $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^{N}$, *the sequence of timed output symbols produced by* ρ *is*

$$s = (\Delta^0, \psi^0), (\Delta^1, \psi^1) \cdots = \{(\Delta^i, \psi^i)\}_{i=0}^{N},$$

where $(\Delta^0, \psi^0) = (0, \iota)$, *and for all* $i \geq 1$, $\Delta^i = \tau^{i-1}$, $\psi^i \in \Psi_o \cup \{\emptyset\}$ *is the output symbol associated with* $e^i \in E$. *For convenience, we define a set of labels* $\Sigma := \mathbb{R}_{\geq 0} \times (\Psi_o \cup \{\emptyset, \iota\})$, *and refer to a sequence of timed output symbols as a label sequence.*

Clearly, each timed event sequence comprises of a time sequence and an event sequence, and each label sequence comprises of a time sequence and an output symbol sequence.

B. Timed Language

Over the set of events E, a set of timed event sequences is called a timed language. Let $\chi(E)$ denote the set of all the timed event sequences that can be generated over $\mathbb{R}_{\geq 0} \times E$. We define a metric on $\chi(E)$ in such a way that the distance only depends on the time sequences if the event sequences of two timed event sequences are the same, and raised to infinity otherwise.

Definition 5 (Timed Event Sequence Metric)**.** *Given* $p_1 = \{e_1^i, \tau_1^i\}_{i=0}^{N_1}, p_2 = \{e_2^i, \tau_2^i\}_{i=0}^{N_2} \in \chi(E)$,

$$d_\chi(p_1, p_2) \triangleq \begin{cases} \sup\limits_i d_{\mathfrak{R}}(\tau_1^i, \tau_2^i) & \text{if } N_1 = N_2, \\ & \text{and } \forall i \in \{0, 1, \ldots, N_1\}, e_1^i = e_2^i; \\ \infty & \text{otherwise.} \end{cases}$$

Definition 6 (Relative Time Metric)**.**

$$d_{\mathfrak{R}}(t_1, t_2) \triangleq \begin{cases} 0 & \text{if } t_1 = t_2 = 0, \\ \frac{|t_1 - t_2|}{t_1 + t_2} & \text{otherwise.} \end{cases}$$

Proposition 1. $d_{\mathfrak{R}}$ *is a metric on* $\mathbb{R}_{\geq 0}$. *The ball* $B(t_1, \epsilon) := \{t | d_{\mathfrak{R}}(t_1, t) \leq \epsilon\}$ *is given by the interval* $[t_1 \frac{1-\epsilon}{1+\epsilon}, t_1 \frac{1+\epsilon}{1-\epsilon}]$.

Proof: Obviously, $d_{\mathfrak{R}}(t_1, t_2) = d_{\mathfrak{R}}(t_2, t_1)$, $d_{\mathfrak{R}}(t_1, t_2) \geq 0$, *and* $d_{\mathfrak{R}}(t_1, t_2) = 0$ *if and only if* $t_1 = t_2$. *Assume* $t_1 \geq t_2 \geq t_3$. *Then*

$$\begin{aligned} & d_{\mathfrak{R}}(t_1, t_2) + d_{\mathfrak{R}}(t_2, t_3) - d_{\mathfrak{R}}(t_1, t_3) \\ = \; & \frac{t_1 - t_2}{t_1 + t_2} + \frac{t_2 - t_3}{t_2 + t_3} - \frac{t_1 - t_3}{t_1 + t_3} \\ = \; & \frac{(t_1 - t_3)(t_1 - t_2)(t_2 - t_3)}{(t_1 + t_2)(t_2 + t_3)(t_1 + t_3)} \\ \geq \; & 0. \end{aligned}$$

Using similar arguments we can prove $d_{\mathfrak{R}}(t_1, t_2) + d_{\mathfrak{R}}(t_1, t_3) \geq d_{\mathfrak{R}}(t_2, t_3)$ *and* $d_{\mathfrak{R}}(t_1, t_3) + d_{\mathfrak{R}}(t_2, t_3) \geq d_{\mathfrak{R}}(t_1, t_2)$.

To compute the ball centered at t_1 *with radius* ϵ:
If $t_2 \geq t_1$, *then* $d_{\mathfrak{R}}(t_1, t_2) \leq \epsilon \Rightarrow \frac{t_2 - t_1}{t_1 + t_2} \leq \epsilon \Rightarrow t_2 \leq t_1 \frac{1+\epsilon}{1-\epsilon}$.
If $t_2 < t_1$, *then* $d_{\mathfrak{R}}(t_1, t_2) \leq \epsilon \Rightarrow \frac{t_1 - t_2}{t_1 + t_2} \leq \epsilon \Rightarrow t_2 \geq t_1 \frac{1-\epsilon}{1+\epsilon}$. ■

This metric is relative in the sense that the distance depends on not only the time difference but also the length of the elapsed time. For example, a 1-second interval and a 2-second one has the same distance as a 100-second interval and a 200-second one. We use the relative metric instead of the more convenient Euclidean metric $|t_1 - t_2|$ because it turns out the latter does not work for the proof of Proposition 5 (Section III-D).

Based on d_χ, a distance measure for timed languages can be defined as the directed or undirected Hausdorff distances:

Definition 7 (Timed Language Metric)**.** *Given* $P_1, P_2 \subset \chi(E)$, *define respectively the directed and undirected Hausdorff distances*

$$\vec{h}(P_1, P_2) \triangleq \sup_{p_1 \in P_1} \inf_{p_2 \in P_2} d_\chi(p_1, p_2),$$

$$h(P_1, P_2) \triangleq \max\{\vec{h}(P_1, P_2), \vec{h}(P_2, P_1)\}.$$

Suppose $\vec{h}(P_1, P_2) \leq \epsilon$ for some $\epsilon \geq 0$. By the definition of Hausdorff distances, for any $s_1 \in P_1$, there exists $s_2 \in P_2$ such that $d_\chi(s_1, s_2) \leq \epsilon$. So we say P_2 approximately includes P_1 (with the precision ϵ). Suppose $h(P_1, P_2) \leq \epsilon$, i.e., P_1, P_2 approximately include each other, then we say P_1, P_2 are approximate equivalent (with the precision ϵ).

C. Abstraction

In this section, we construct a system abstraction that is useful for fault diagnosability analysis and diagnoser construction of H. The abstraction consists of finitely many simulated trajectories. To that end, we make extensive use of results reported in [26], [27], whose details are not presented in this paper because of space limitation.

First, randomly take a point $(\hat{\ell}^0, \hat{x}^0) \in L^0 \times X^0$ and simulate a trajectory $\hat{\rho} = \{(\hat{e}^i, \hat{\ell}^i, \hat{x}^i, \hat{\tau}^i)\}_{i=0}^{\hat{N}}$ for finite time horizon $[0, t_{end}]$ (or for $\hat{N}$ events triggered). Based on $\hat{\rho}$, the algorithm in [26] computes a robust neighborhood around $(\hat{\ell}^0, \hat{x}^0)$, denoted as $Robust(\hat{\ell}^0, \hat{x}^0)$. The robust neighborhood computed with the parameter ϵ has the following property [26]:

Proposition 2. *For any* $(\ell^0, x^0) \in Robust(\hat{\ell}^0, \hat{x}^0)$, *for any trajectory* $\rho' = \{(e'^i, \ell'^i, x'^i, \tau'^i)\}_{i=0}^{N'}$ *initiated from* (ℓ^0, x^0), *there exists a trajectory* $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^{N}$ *such that* ρ *is a sub-trajectory of* ρ' *or* ρ' *is a sub-trajectory of* ρ, *and* $\rho, \hat{\rho}$ *satisfy*

- $N = \hat{N}$, $e^i = \hat{e}^i$, $d_{\mathfrak{R}}(\tau^i, \hat{\tau}^i) \leq \epsilon$ *for all* $i \in [0, N]$.

By Definition 2, when ρ is a sub-trajectory of ρ', $N \leq N'$, $(e^i, \ell^i, x^i, \tau^i) = (e'^i, \ell'^i, x'^i, \tau'^i)$ for all $i \in [0, N-1]$, $(e^N, \ell^N, x^N) = (e'^N, \ell'^N, x'^N)$, $\tau^N \leq \tau'^N$, and thus $\sum_{i=0}^{N} \tau^i < \sum_{i=0}^{N'} \tau'^i$.

Note that if the trajectories initiated from $(\hat{\ell}^0, \hat{x}^0)$ trigger multiple timed event sequences (see the non-determinism assumption in Section II-A), such robust neighborhood might not exist. We thus need to use the adapted approach presented in [27] to compute robust neighborhoods (called safe neighborhoods in [27] because of the application to system safety verification) around such an initial state. Instead of a single trajectory $\hat{\rho}$, a finite set of representative trajectories $[\hat{\rho}]$ are simulated from $(\hat{\ell}^0, \hat{x}^0)$ (triggering multiple timed event sequences), and used to compute $Robust(\hat{\ell}^0, \hat{x}^0)$. As a result, for any $(\ell^0, x^0) \in Robust(\hat{\ell}^0, \hat{x}^0)$, for any trajectory $\rho' = \{(e'^i, \ell'^i, x'^i, \tau'^i)\}_{i=0}^{N'}$ initiated from (ℓ^0, x^0), there exists $\hat{\rho} = \{(\hat{e}^i, \hat{\ell}^i, \hat{x}^i, \hat{\tau}^i)\}_{i=0}^{\hat{N}} \in [\hat{\rho}]$, and a trajectory $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^{N}$, such that the properties listed in Proposition 2 hold.

Next, we compute the robust neighborhoods around more simulated initial states, and try to fully cover $L^0 \times X^0$ with the union of these neighborhoods. The robust neighborhoods computed with the algorithm in [27] are not empty even for an initial state that possesses non-deterministic timed event sequences. However, the radii of $\{Robust(\hat{\ell}^0, \hat{x}^0) | (\hat{\ell}^0, \hat{x}^0) \in L^0 \times X^0\}$ are not bounded from below by a positive number. Consequently, randomly generating

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

robust neighborhoods and covering $L^0 \times X^0$ may never terminate with 100% coverage.

This problem is fixed by setting the threshold value d_{thr} in [27] to a positive number instead of 0. Then for each $(\hat{\ell}^0, \hat{x}^0)$ we obtain $[\hat{\rho}]^e$, a finite set of representative (virtual) trajectories simulated from $(\hat{\ell}^0, \hat{x}^0)$. In the process of simulating $[\hat{\rho}]^e$, an event can be triggered even if the corresponding guard is not reached by the continuous system state (but the system state should be within d_{thr} distance to the guard). Therefore, some trajectories non-existent according to the original discrete dynamics are included in $[\hat{\rho}]^e$. These are called *virtual trajectories* [27].

Definition 8 (Virtual Trajectory)**.** *Given $H = (L \times X, L^0 \times X^0, D, E, Inv)$, a metric $\phi : X \times X \to \mathbb{R}_{\geq 0}$, and $d_{thr} \geq 0$, a virtual trajectory of H is*

$$\rho = (e^0, \ell^0, x^0, \tau^0), (e^1, \ell^1, x^1, \tau^1) \cdots = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^N,$$

such that

- *for all $i \geq 0$, $(\ell^i, x^i) \in L \times X$, and $(\ell^0, x^0) \in L^0 \times X^0$;*
- *for all $i \geq 0$, $\tau^i \in \mathbb{R}_{\geq 0}$, and $\xi_{\ell^i}(t, x^i) \in Inv(\ell^i)$ for all $t \in [0, \tau^i]$;*
- *for all $i \geq 1$, $e^i = (\ell^{i-1}, \ell^i, g^i, r^i) \in E$, (ℓ^i, x^i) is the reset state for e^i, where $x^i = r^i(y^i)$ for some $y^i \in g^i$, and $\phi(\xi_{\ell^{i-1}}(\tau^{i-1}, x^{i-1}), y^i) \leq d_{thr}$.*

Sub-trajectories can be defined in the same manner as Definition 2.

With $[\hat{\rho}]^e$ simulated, the same algorithm in [27] can be used to compute the so-called enlarged neighborhoods $Robust^e(\hat{\ell}^0, \hat{x}^0)$. As a result, for any $(\ell^0, x^0) \in Robust^e(\hat{\ell}^0, \hat{x}^0)$, for any trajectory $\rho' = \{(e'^i, \ell'^i, x'^i, \tau'^i)\}_{i=0}^{N'}$ initiated from (ℓ^0, x^0), there exist $\hat{\rho} = \{(\hat{e}^0, \hat{\ell}^i, \hat{x}^i, \hat{\tau}^i)\}_{i=0}^{\hat{N}} \in [\hat{\rho}]^e$, and a trajectory $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^N$, such that the properties listed in Prop. 2 hold. Moreover, the radii of $\{Robust^e(\hat{\ell}^0, \hat{x}^0) | (\hat{\ell}^0, \hat{x}^0) \in L^0 \times X^0\}$ are bounded from below by a positive number [27]. Therefore, $L^0 \times X^0$ can be fully covered by finitely many neighborhoods that are randomly generated. We denote the simulated initial states as $\hat{L}^0 \times \hat{X}^0 \subset L^0 \times X^0$, whose enlarged robust neighborhoods (or robust neighborhoods if possible) cover $L^0 \times X^0$.

The system abstraction is constructed with the finitely many (virtual) trajectories simulated in the construction of $\hat{L}^0 \times \hat{X}^0$. We use $\hat{\mathcal{J}}(H)$ to denote the set of sub-trajectories of these simulated (virtual) trajectories such that the trajectory horizon does not exceed the specified horizon. By definition, sub-trajectories include the trajectory itself. Depending on the whether the horizon is specified by the total dwell time $\hat{t}_{max}$ or the number of events triggered $\hat{N}_{max}$, $\hat{\mathcal{J}}(H)$ can be $\hat{\mathcal{J}}_R(\hat{t}_{max}, H)$ or $\hat{\mathcal{J}}_N(\hat{N}_{max}, H)$.

- $\hat{\mathcal{J}}_R(\hat{t}_{max}, H)$ is defined to be the set of all the sub-trajectories $\hat{\rho} = \{(\hat{e}^i, \hat{\ell}^i, \hat{x}^i, \hat{\tau}^i)\}_{i=0}^{\hat{N}}$ of the simulated (virtual) trajectories such that $\sum_{i=0}^{\hat{N}} \hat{\tau}^i \leq \hat{t}_{max}$;
- $\hat{\mathcal{J}}_N(\hat{N}_{max}, H)$ is defined to be the set of all the sub-trajectories $\hat{\rho} = \{(\hat{e}^i, \hat{\ell}^i, \hat{x}^i, \hat{\tau}^i)\}_{i=0}^{\hat{N}}$ of the simulated (virtual) trajectories such that $\hat{N} \leq \hat{N}_{max}$.

In what follows, we use $\mathcal{J}(H)$ to denote the trajectories of H that are of interest in the finite-horizon fault diagnosis. $\mathcal{J}(H)$ can be specified as $\mathcal{J}_R(t_{max}, H)$ or $\mathcal{J}_N(N_{max}, H)$:

- $\mathcal{J}_R(t_{max}, H)$ is defined to be the set of all the trajectories $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^N$ of H such that $\sum_{i=0}^{N} \tau^i \leq t_{max}$;
- $\mathcal{J}_N(N_{max}, H)$ is defined to be the set of all the trajectories $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^N$ of H such that $N \leq N_{max}$.

Let $\mathcal{P}_R(t_{max}, H), \mathcal{P}_N(N_{max}, H), \hat{\mathcal{P}}_R(\hat{t}_{max}, H), \hat{\mathcal{P}}_N(\hat{N}_{max}, H)$ be respectively the set of timed event sequences produced by $\mathcal{J}_R(t_{max}, H), \mathcal{J}_N(N_{max}, H), \hat{\mathcal{J}}_R(\hat{t}_{max}, H), \hat{\mathcal{J}}_N(\hat{N}_{max}, H)$. Then the following proposition holds:

Proposition 3. *Given the simulation horizon $[0, t_{end}]$ (or $\hat{N}$) in the construction of $\hat{L}^0 \times \hat{X}^0$,*

1) *if $\hat{L}^0 \times \hat{X}^0$ is constructed from robust neighborhoods, then we have*

$$\vec{h}(\mathcal{P}_R(t_{end}\frac{1-\epsilon}{1+\epsilon}, H), \hat{\mathcal{P}}_R(t_{end}, H)) \leq \epsilon, \quad (1)$$

$$\vec{h}(\mathcal{P}_N(\hat{N}-1, H), \hat{\mathcal{P}}_N(\hat{N}-1, H)) \leq \epsilon, \quad (2)$$

$$\vec{h}(\hat{\mathcal{P}}_R(t_{end}, H), \mathcal{P}_R(t_{end}, H)) = 0, \quad (3)$$

$$\vec{h}(\hat{\mathcal{P}}_N(\hat{N}-1, H), \mathcal{P}_N(\hat{N}-1, H)) = 0; \quad (4)$$

2) *if $\hat{L}^0 \times \hat{X}^0$ is constructed from enlarged robust neighborhoods, then we have Eq.* (1)(2)*.*

Proof: Suppose the simulation horizon for $\hat{L}^0 \times \hat{X}^0$ is specified by the total dwell time t_{end}. For any timed event sequence $p' \in \mathcal{P}_R(t_{end}\frac{1-\epsilon}{1+\epsilon}, H)$, by definition, there is a trajectory $\rho' = \{(e'^i, \ell'^i, x'^i, \tau'^i)\}_{i=0}^{N'}$ initiated from $(\ell^0, x^0) \in L^0 \times X^0$ that produces p' and satisfies $\sum_{i=0}^{N'} \tau'^i \leq t_{end}\frac{1-\epsilon}{1+\epsilon}$. Let $(\hat{\ell}^0, \hat{x}^0) \in \hat{L}^0 \times \hat{X}^0$ be a simulated initial state whose (enlarged) robust neighborhood covers (ℓ^0, x^0). By the property of (enlarged) robust neighborhood, there exist $\hat{\rho} = \{(\hat{e}^i, \hat{\ell}^i, \hat{x}^i, \hat{\tau}^i)\}_{i=0}^{\hat{N}}$ and $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^N$ such that $N = \hat{N}$, $e^i = \hat{e}^i$, $d_{\mathfrak{R}}(\tau^i, \hat{\tau}^i) \leq \epsilon$ for all $i \in [0, N]$, where $\hat{\rho}$ is simulated from $(\hat{\ell}^0, \hat{x}^0)$ for the specified horizon, and ρ is a sub-trajectory of ρ' or the other way around.

By Proposition 1, $\tau^i \in [\hat{\tau}^i\frac{1-\epsilon}{1+\epsilon}, \hat{\tau}^i\frac{1+\epsilon}{1-\epsilon}]$ for all $i \in [0, N]$, which implies

$$\Sigma_{i=0}^N \tau^i \in [\Sigma_{i=0}^N \hat{\tau}^i \frac{1-\epsilon}{1+\epsilon}, \Sigma_{i=0}^N \hat{\tau}^i \frac{1+\epsilon}{1-\epsilon}] = [t_{end}\frac{1-\epsilon}{1+\epsilon}, t_{end}\frac{1+\epsilon}{1-\epsilon}].$$

It follows from $\sum_{i=0}^{N'} \tau'^i \leq t_{end}\frac{1-\epsilon}{1+\epsilon}$ that $\sum_{i=0}^{N} \tau^i \geq \sum_{i=0}^{N'} \tau'^i$. Thus, ρ' is a sub-trajectory of ρ. In specific, $N' \leq N$, $(e^i, \ell^i, x^i, \tau^i) = (e'^i, \ell'^i, x'^i, \tau'^i)$ for all $i \in [0, N'-1]$, $(e^{N'}, \ell^{N'}, x^{N'}) = (e'^{N'}, \ell'^{N'}, x'^{N'})$, $\tau^{N'} \geq \tau'^{N'}$. Since $d_{\mathfrak{R}}(\tau^{N'}, \hat{\tau}^{N'}) \leq \epsilon, \tau^{N'} \geq \tau'^{N'}$, there exists $\tau_1 \in [0, \hat{\tau}^{N'}]$ such that $d_{\mathfrak{R}}(\tau'^{N'}, \tau_1) \leq \epsilon$. It follows that $\hat{\rho}$ has a sub-trajectory $\hat{\rho}' = \{(\hat{e}'^i, \hat{\ell}'^i, \hat{x}'^i, \hat{\tau}'^i)\}_{i=0}^{\hat{N}'}$, such that $\hat{N}' = N'$, $\hat{e}'^i = e'^i, d_{\mathfrak{R}}(\hat{\tau}'^i, \tau'^i) \leq \epsilon$ for all $i \in [0, N']$. Therefore, the timed event sequences $\hat{p}', p'$ respectively produced by $\hat{\rho}', \rho'$ satisfy that $d_\chi(\hat{p}', p') \leq \epsilon$. Since $\hat{p}' \in \hat{\mathcal{P}}_R(t_{end}, H)$, which follows from $\hat{p} \in \hat{\mathcal{P}}_R(t_{end}, H)$, we have $\vec{h}(\mathcal{P}_R(t_{end}\frac{1-\epsilon}{1+\epsilon}, H), \hat{\mathcal{P}}_R(t_{end}, H)) \leq \epsilon$.

If $\hat{L}^0 \times \hat{X}^0$ is constructed from robust neighborhoods, then for any timed event sequence $\hat{p} \in \hat{\mathcal{P}}_R(t_{end}, H)$, clearly $\hat{p} \in \mathcal{P}_R(t_{end}, H)$ holds. Thus, $\vec{h}(\hat{\mathcal{P}}_R(t_{end}, H), \mathcal{P}_R(t_{end}, H)) = 0$.

Suppose the simulation horizon for $\hat{L}^0 \times \hat{X}^0$ is specified by the number of triggered events $\hat{N}$. By using similar argument as before, for any $p' = \{(e'^i, \tau'^i)\}_{i=0}^{N'} \in \mathcal{P}_N(\hat{N}-1, H)$, there exists $\hat{p}' = \{(\hat{e}'^i, \hat{\tau}'^i)\}_{i=0}^{\hat{N}'} \in \hat{\mathcal{P}}_N(\hat{N}-1, H)$, such that $N' = \hat{N}'$ and $e'^i = \hat{e}'^i$, $d_{\mathfrak{R}}(\hat{\tau}'^i, \tau'^i) \leq \epsilon$ for all $i \in [0, N']$. Therefore, $\vec{h}(\mathcal{P}_N(\hat{N}-1, H), \hat{\mathcal{P}}_N(\hat{N}-1, H)) \leq \epsilon$.

If $\hat{L}^0 \times \hat{X}^0$ is constructed from robust neighborhoods, then for any $\hat{p} \in \hat{\mathcal{P}}_N(\hat{N}-1, H)$, clearly $\hat{p} \in \mathcal{P}_N(\hat{N}-1, H)$ holds. Thus, $\vec{h}(\hat{\mathcal{P}}_N(\hat{N}-1, H), \mathcal{P}_N(\hat{N}-1, H)) = 0$. ■

If $\hat{L}^0 \times \hat{X}^0$ is constructed from enlarged robust neighborhoods, then we cannot prove Eq. (3)(4). Due to the virtual trajectories in $[\hat{\rho}]^e$ simulated from $\hat{L}^0 \times \hat{X}^0$, we only obtain approximate timed language inclusion in one direction, i.e., the timed language of the original system is approximately included by that of the abstraction, but not the other way around. Suppose the following condition is satisfied:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

- for all $(\hat{\ell}^0, \hat{x}^0) \in \hat{L}^0 \times \hat{X}^0$, for all $\hat{\rho} \in [\hat{\rho}]^e$, H has a trajectory ρ such that $d_\chi(\hat{\rho}, p) \leq \epsilon$.

Then instead of Eq. (3)(4), the following holds:

$$\vec{h}(\hat{\mathcal{P}}_R(t_{end}, H), \mathcal{P}_R(t_{end}\frac{1+\epsilon}{1-\epsilon}, H)) \leq \epsilon, \quad (5)$$

$$\vec{h}(\hat{\mathcal{P}}_N(\hat{N}-1, H), \mathcal{P}_N(\hat{N}-1, H)) \leq \epsilon. \quad (6)$$

In this case, we still obtain approximate timed language equivalence. It can be proved that when the positive threshold value d_{thr} in the algorithm of [27] is sufficiently small, such condition can be satisfied; but for the present work to be focused, we do not discuss the proof.

The abstraction has finitely many trajectories that extend to the end of the simulation horizon. Its diagnosability can be easily analyzed and used to derive the diagnosability of H. We see this in the next section.

III. Diagnosability with Measurement Uncertainty

A. Projected Label Sequences

In Section III, we investigate the problem of diagnosing faults for hybrid automata without directly observing the trajectories.

During the system operation, one can only observe a sequence of labels, which is a sequence of timed output symbols; but due to the unobservable events, this observed label sequence may be different from the original one produced by the trajectory. For example, listening to different sounds with time intervals generated by a machine would provide the information for fault diagnosis; but not every sound is audible, and sometimes an unobservable output symbol $\emptyset$ occurs. With this setting, we introduce the definition of projected label sequences in this section.

Definition 9 (Projected Label Sequence)**.** *Let Σ^* denote the set of all the label sequences generated over Σ. Let $s = \{(\Delta^i, \psi^i)\}_{i=0}^N \in \Sigma^*$ be a label sequence, and $\Pi : \Sigma^* \to \Sigma^*$ be a single-valued projection map. Then $\pi := \Pi(s)$ is called the projected label sequence of s through the map Π.*

We define the projection map Π that absorbs all the labels with the unobservable output symbol $\emptyset$ into the first observable label that follows, while leaves the rest of labels unchanged. That is, Π projects s to π in the sense that every unobservable output symbol $\emptyset$ is erased, whose dwell time gets added into the dwell time of the next label that has an observable output symbol. For instance, $(\Delta^0, \psi^0), (\Delta^1, \emptyset), (\Delta^2, \psi^2)$ is projected to $(\Delta^0, \psi^0), (\Delta^1 + \Delta^2, \psi^2)$. If a trajectory has consecutive unobservable output symbols at its end, then the unobservable end is abandoned in the projected label sequence. Projected label sequences are the only accessible information for system diagnosis. They contain two aspects of information as below, where (Δ^i, ψ^i) are the labels after projection:

1) Before the starting signal $\psi^0 = \iota$, the system operation is clear of faults.
2) Each observable symbol $\psi^i, i \geq 1$ should be output Δ^i time units later than the preceding one; in the meantime, no observable symbol can be output.

In what follows, a metric on Σ^* is defined in such a way that the distance only depends on the time sequences if the output symbol sequences of two label sequences are the same, and raised to infinity otherwise. This is motivated by the application of diagnosing faults by observing the projected label sequences. Unlike dwell time, which may be measured with uncertainty, different output symbols are assumed to be readily differentiable from each other.

Definition 10 (Label Sequence Metric)**.** *Given $s_1 = \{\Delta_1^i, \psi_1^i\}_{i=0}^{N_1}, s_2 = \{\Delta_2^i, \psi_2^i\}_{i=0}^{N_2} \in \Sigma^*$,*

$$d_{\Sigma^*}(s_1, s_2) \triangleq \begin{cases} \sup_i d_{\Re}(\Delta_1^i, \Delta_2^i) & \text{if } N_1 = N_2, \\ & \text{and } \forall i \in [0, N_1], \psi_1^i = \psi_2^i; \\ \infty & \text{otherwise;} \end{cases}$$

where $d_{\Re}$ is the relative time metric defined in Section II-B.

B. Definition of (δ_d, δ_m)-Diagnosability

In fault diagnosis two main tasks are fault detection and isolation. Detecting a fault requires its manifestation being discriminable from that of normal system behaviors, while isolating requires further discriminability from symptoms of other faults. To investigate the diagnosability we start from defining the discriminability of a fault from normal behaviors and other faults based on the system model.

Consider a hybrid automaton $H = (L \times X, L^0 \times X^0, D, E, Inv)$, and the projection map Π defined in Section III-A. Let $L^f \subset L$ be the set of locations that model a failure: L^f is called the faulty set, whose elements can be partitioned into M disjoint subsets $\bigcup_{j=1}^{M} L_j^f = L^f$. Each faulty subset L_j^f corresponds to a type of fault $F_j \in \mathbb{F} := \{F_1, \ldots, F_M\}$. The discriminability of a fault F_j is proposed below, which takes into account the delay in discriminating a fault since it first occurs, as well as the measurement uncertainty of time intervals.

Definition 11 ($F_j\delta$-Faulty Trajectory)**.** *A trajectory $\rho = \{(e^i, \ell^i, x^i, \tau^i)\}_{i=0}^N$ is $F_j\delta$-faulty if and only if there exists a finite index $i^f \in [0, N]$ such that:*

1) *$\forall i < i^f$, $\ell^i \notin L_j^f$;*
2) *$\ell^{i^f} \in L_j^f$;*
3) *$\sum_{i=i^f}^{N} \tau^i \geq \delta$.*

In the special case where $\delta = 0$, i.e., only the occurrence of F_j is required, the term F_j-faulty is used. If a trajectory is free of any fault, then it is called normal.

Definition 12 ((δ_d, δ_m)-Discriminability)**.** *Let J be a set of trajectories, ρ_1, ρ_2 be two arbitrary trajectories in J, which produce the label sequences s_1, s_2, and $\pi_1 = \Pi(s_1), \pi_2 = \Pi(s_2)$.*

1) *F_j is called (δ_d, δ_m)-discriminable from normal (with respect to J) if and only if the following is satisfied: if ρ_1 is a $F_j\delta_d$-faulty trajectory, then either ρ_2 is not normal or $d_{\Sigma^*}(\pi_1, \pi_2) > \delta_m$.*
2) *Given $F_i \in \mathbb{F} = \{F_1, \cdots, F_M\}$, F_j is called (δ_d, δ_m)-discriminable from F_i (with respect to J) if and only if the following is satisfied: if ρ_1 is a $F_j\delta_d$-faulty trajectory, then either ρ_2 is not F_i-faulty or $d_{\Sigma^*}(\pi_1, \pi_2) > \delta_m$.*

In what follows, we define the diagnosability based on the (δ_d, δ_m)-discriminability of faults. We want to perform finite-horizon diagnosability analysis of H. To that end, we need to consider the discriminability of faults with respect to $\mathcal{J}(H)$. As mentioned before, $\mathcal{J}(H)$ can be specified specified as $\mathcal{J}_R(t_{max}, H)$ (or $\mathcal{J}_N(N_{max}, H)$), i.e., the set of all the possible trajectories of H such that the trajectory horizon does not exceed t_{max} (or N_{max}). We also need to consider the discriminability of faults with respect to $\hat{\mathcal{J}}_R(\hat{t}_{max}, H)$ (or $\hat{\mathcal{J}}_N(\hat{N}_{max}, H)$), since the diagnosability analysis of the original system has to be performed indirectly from the system abstraction.

Essentially, $d_{\Sigma^*}(\pi_1, \pi_2) > \delta_m$ requires that a pair of projected label sequences coming from a $F_j\delta_d$-faulty trajectory and a normal/F_i-faulty trajectory can be distinguished from each other,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

given the measurement uncertainty δ_m for time intervals. Thus, (δ_d, δ_m)-discriminability from normal amounts to the detectability of F_j with the delay δ_d and measurement uncertainty δ_m. As for isolability, things are more complicated, since F_j has to be (δ_d, δ_m)-discriminable from all other faults, namely all $F_i \in \mathbb{F} = \{F_1, \cdots, F_M\}$ such that $F_i \neq F_j$. The semantics of isolating F_j from F_i is to exclude the occurrence of F_i as a cause to alarm, relying on the possibly delayed symptoms of F_j.

Another remark about the (δ_d, δ_m)-discriminability of F_j from F_i is on asymmetry. To see this, suppose ρ_1, ρ_2 are too close to discriminate in the sense that $d_{\Sigma^*}(\pi_1, \pi_2) \leq \delta_m$. If ρ_1 is $F_j\delta_d$-faulty and ρ_2 is $F_i\delta'_d$-faulty with $\delta'_d < \delta_d$, then F_j is not (δ_d, δ_m)-discriminable from F_i, but F_i can still be (δ_d, δ_m)-discriminable from F_j.

Definition 13 ((δ_d, δ_m)-Diagnosability)**.** *Given a set of trajectories J and a faulty set $L^f = \{L^f_1, \cdots, L^f_M\}$, we have the following definitions with respect to J:*

- *$F_j \in \mathbb{F} = \{F_1, \cdots, F_M\}$ is (δ_d, δ_m)-detectable if and only if it is (δ_d, δ_m)-discriminable from normal;*
- *F_j is (δ_d, δ_m)-isolable if and only if it is (δ_d, δ_m)-discriminable from all other faults $F_i \in \mathbb{F}, F_i \neq F_j$;*
- *F_j is (δ_d, δ_m)-diagnosable if and only if it is (δ_d, δ_m)-detectable and isolable.*

The set of trajectories J is said to be (δ_d, δ_m)-diagnosable if and only if all the faults in $\mathbb{F}$ are (δ_d, δ_m)-diagnosable with respect to J. In particular, if $\mathcal{J}(H)$ is (δ_d, δ_m)-diagnosable, we also say that the system H is (δ_d, δ_m)-diagnosable (for the considered horizon).

In discussions above the discriminability of faults under consideration is confined to single fault. This can be easily extended to multiple-fault cases, since the discriminablity is defined based on the $F_j\delta$-faulty trajectory definition, which is readily modifiable by using *and* operators. For example, given $\mathbb{F} = \{F_1, F_2, F_3\}$ and a combination of faults $f = \{F_1, F_3\} \in 2^{\mathbb{F}}$, define $f\delta$-faulty trajectories as both $F_1\delta$-faulty and $F_3\delta$-faulty, i.e., the intersection of the sets of $F_1\delta$-faulty and $F_3\delta$-faulty trajectories forms the set of $f\delta$-faulty trajectories. Then multiple-fault diagnosability can be defined and analyzed. For simplicity we consider the single-fault case.

Proposition 4. *Given H and L^f, the following statements hold:*

1) *If H is (δ_d, δ_m)-diagnosable, then it is (δ^*_d, δ_m)-diagnosable for all $\delta^*_d \geq \delta_d$.*
2) *If H is not (δ_d, δ_m)-diagnosable, then it is not (δ^*_d, δ_m)-diagnosable for all $\delta^*_d \leq \delta_d$.*
3) *If H is (δ_d, δ_m)-diagnosable, then it is (δ_d, δ^*_m)-diagnosable for all $\delta^*_m \leq \delta_m$.*
4) *If H is not (δ_d, δ_m)-diagnosable, then it is not (δ^*_d, δ_m)-diagnosable for all $\delta^*_m \geq \delta_m$.*

Proof: Straightforward by Definition 13. ∎

C. Comparison of Diagnosability Notions

The classical DES diagnosability notion [10] relies on event sequences. Extended to a multiple-fault version in [28], it states that the DES is diagnosable if and only if for all faults $F_j \in \mathbb{F}$, all F_j-faulty trajectories can be determined via the projections of prefixes whose length are uniformly bounded, that is, discriminated from trajectories that are not F_j-faulty. This uniform bound on event numbers gives the maximum delay for diagnosing F_j since it first occurs, but the number is not prescribed explicitly in the definition.

The diagnosability definition for timed automata proposed by [12] is different from the DES diagnosability in some ways: Dense time instead of discrete time (counting events) is considered. The F_j-faulty and non-F_j-faulty trajectories (which refer to trajectories that are not F_j-faulty) are also required to be discriminable within a maximum delay by projections; but the maximum delay is explicitly prescribed in the notion of Δ-diagnosability. So the fault F_j has to be determined at most Δ (integral) time units later than its first occurrence. If the system is Δ-diagnosable for some natural number Δ, then it is called diagnosable.

Similar to the Δ-diagnosability for timed automata, the (δ_d, δ_m)-diagnosability for hybrid automata prescribes a maximum delay δ_d in (real) time units as the diagnosis time window. So the delays Δ and δ_d are basically the same. The (δ_d, δ_m)-diagnosability additionally conveys the uncertainty δ_m in symptom measurements. In specific, a metric d_{Σ^*} has been defined to compute the distance between the projected label sequences, and a distance greater than the measurement uncertainty δ_m ensures the discriminability. By contrast, direct discrimination of different projected label sequences as in the DES and timed automata diagnosability corresponds to the case $\delta_m = 0$.

Moreover, the (δ_d, δ_m)-diagnosability differs in that it requires the discriminability of $F_j\delta_d$-faulty trajectories from normal trajectories and other faulty trajectories rather than from non-F_j-faulty trajectories. Note that discriminability from non-F_j-faulty trajectories does not exclude other faults as the causes to alarms (for instance, a trajectory is both F_j-faulty and F_i-faulty), although it implicitly excludes normality. As for the (δ_d, δ_m)-diagnosability, both detectability and isolability are involved.

D. Bridging The Timed Language to (δ_d, δ_m)-Diagnosability

The projected label sequences are the information available for fault diagnosis, while properties of the timed languages are what we can get from system abstraction. Thus a relation should be established between the metrics for projected label sequences and timed event sequences.

Proposition 5. *Given $s_1, s_2 \in \Sigma^*$ as the label sequences produced by the timed event sequences $p_1, p_2 \in \chi(E)$, and $\pi_1 = \Pi(s_1), \pi_2 = \Pi(s_2)$, we have $d_{\Sigma^*}(\pi_1, \pi_2) \leq d_\chi(p_1, p_2)$.*

Proof: If $d_\chi(p_1, p_2)$ is finite, then clearly $d_{\Sigma^*}(s_1, s_2) = d_\chi(p_1, p_2)$, and s_1, s_2 have the same output symbol sequence. According to the definition of Π, π_1, π_2 must also have the same output symbol sequence. Moreover, for any $\Delta^1_1, \Delta^1_2, \Delta^2_1, \Delta^2_2, > 0$, $\frac{|\Delta^1_1+\Delta^2_1-\Delta^1_2-\Delta^2_2|}{\Delta^1_1+\Delta^2_1+\Delta^1_2+\Delta^2_2} \leq \frac{|\Delta^1_1-\Delta^1_2|+|\Delta^2_1-\Delta^2_2|}{\Delta^1_1+\Delta^2_1+\Delta^1_2+\Delta^2_2} \leq \max\{\frac{|\Delta^1_1-\Delta^1_2|}{\Delta^1_1+\Delta^1_2}, \frac{|\Delta^2_1-\Delta^2_2|}{\Delta^2_1+\Delta^2_2}\}$. Therefore, by combining some labels' dwell time distance after the projection, the supremum of dwell time distance over all the labels has become smaller. Thus, we obtain $d_{\Sigma^*}(\pi_1, \pi_2) \leq d_{\Sigma^*}(s_1, s_2)$. ∎

For two sets of trajectories $\mathcal{J}(H), \hat{\mathcal{J}}(H)$ satisfying $\vec{h}(\mathcal{P}(H), \hat{\mathcal{P}}(H)) \leq \epsilon$, we now show there is a relation between their (δ_d, δ_m)-diagnosability.

Lemma 1. *Given a faulty set L^f, and $\mathcal{J}(H), \hat{\mathcal{J}}(H)$ that satisfy $\vec{h}(\mathcal{P}(H), \hat{\mathcal{P}}(H)) \leq \epsilon$, if $\mathcal{J}(H)$ is not (δ_d, δ_m)-diagnosable, then $\hat{\mathcal{J}}(H)$ is not $(\delta_d \frac{1-\epsilon}{1+\epsilon}, \delta_m + 2\epsilon)$-diagnosable.*

Proof: For convenience, let $\mathcal{F}_j(H) \subset \mathcal{J}(H), \hat{\mathcal{F}}_j(H) \subset \hat{\mathcal{J}}(H)$ denote the set of F_j-faulty trajectories, and $\mathcal{F}^\delta_j(H) \subset \mathcal{F}_j(H), \hat{\mathcal{F}}^\delta_j(H) \subset \hat{\mathcal{F}}_j(H)$ denote the set of $F_j\delta$-faulty trajectories.

Suppose $\mathcal{J}(H)$ is not (δ_d, δ_m)-diagnosable. This implies, by Definition 12 and 13, the existence of a fault F_j and two trajectories $\rho_1, \rho_2 \in \mathcal{J}(H)$,

$$\begin{aligned} \rho_1 &= (e^0_1, \ell^0_1, x^0_1, \tau^0_1), \ldots, (e^{i^f}_1, \ell^{i^f}_1, x^{i^f}_1, \tau^{i^f}_1), \\ &\quad \ldots, (e^{N_1}_1, \ell^{N_1}_1, x^{N_1}_1, \tau^{N_1}_1), \\ \rho_2 &= (e^0_2, \ell^0_2, x^0_2, \tau^0_2), \ldots, (e^{N_2}_2, \ell^{N_2}_2, x^{N_2}_2, \tau^{N_2}_2), \end{aligned}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

such that $\rho_1 \in \mathcal{F}_j^{\delta_d}(H)$, ρ_2 is either normal or contained in $\mathcal{F}_i(H)$ for some fault $F_i \neq F_j$, and their projected label sequences satisfy $d_{\Sigma^*}(\pi_1, \pi_2) \leq \delta_m$.

The timed event sequences produced by ρ_1, ρ_2 are in $\mathcal{P}(H)$ by definition. Denote them as $p_1 = \{(e_1^i, \tau_1^i)\}_{i=0}^{N_1}, p_2 = \{(e_2^i, \tau_2^i)\}_{i=0}^{N_2}$ respectively. Since $\rho_1 \in \mathcal{F}_j^{\delta_d}(H)$, then $\sum_{i=i^f}^{N_1} \tau_1^i \geq \delta_d$.

Since $\vec{h}(\mathcal{P}(H), \hat{\mathcal{P}}(H)) \leq \epsilon$, there exist two trajectories $\hat{\rho}_1, \hat{\rho}_2 \in \hat{\mathcal{J}}(H)$,

$$\begin{aligned}\hat{\rho}_1 &= (\hat{e}_1^0, \hat{\ell}_1^0, \hat{x}_1^0, \hat{\tau}_1^0), \ldots, (\hat{e}_1^{i^f}, \hat{\ell}_1^{i^f}, \hat{x}_1^{i^f}, \hat{\tau}_1^{i^f}),\\ &\quad \ldots, (\hat{e}_1^{\hat{N}_1}, \hat{\ell}_1^{\hat{N}_1}, \hat{x}_1^{\hat{N}_1}, \hat{\tau}_1^{\hat{N}_1}),\\ \hat{\rho}_2 &= (\hat{e}_2^0, \hat{\ell}_2^0, \hat{x}_2^0, \hat{\tau}_2^0), \ldots, (\hat{e}_2^{\hat{N}_2}, \hat{\ell}_2^{\hat{N}_2}, \hat{x}_2^{\hat{N}_2}, \hat{\tau}_2^{\hat{N}_2}),\end{aligned}$$

such that their timed event sequences $\hat{p}_1 = \{(\hat{e}_1^i, \hat{\tau}_1^i)\}_{i=0}^{\hat{N}_1}, \hat{p}_2 = \{(\hat{e}_2^i, \hat{\tau}_2^i)\}_{i=0}^{\hat{N}_2}$ satisfy

$$d_\chi(p_1, \hat{p}_1) \leq \epsilon, d_\chi(p_2, \hat{p}_2) \leq \epsilon. \tag{7}$$

Clearly, p_1 and $\hat{p}_1$ have the same event sequence, p_2 and $\hat{p}_2$ have the same event sequence. It follows that $\hat{\rho}_1 \in \hat{\mathcal{F}}_j(H)$, $\hat{\rho}_2$ is either normal or contained in $\hat{\mathcal{F}}_i(H)$.

Let $\pi_1, \pi_2, \hat{\pi}_1, \hat{\pi}_2$ be the projected label sequences of $\rho_1, \rho_2, \hat{\rho}_1, \hat{\rho}_2$. By the triangle inequality, Prop. 5 and Eq. (7), the following holds:

$$\begin{aligned}d_{\Sigma^*}(\hat{\pi}_1, \hat{\pi}_2) &\leq d_{\Sigma^*}(\pi_1, \hat{\pi}_1) + d_{\Sigma^*}(\pi_1, \pi_2) + d_{\Sigma^*}(\pi_2, \hat{\pi}_2)\\ &\leq \delta_m + 2\epsilon.\end{aligned}$$

From Eq. (7) we have that for all $i \in [0, N_1]$, $d_{\Re}(\tau_1^i, \hat{\tau}_1^i) \leq \epsilon$, namely, $\tau_1^i \frac{1-\epsilon}{1+\epsilon} \leq \hat{\tau}_1^i \leq \tau_1^i \frac{1+\epsilon}{1-\epsilon}$. It follows that

$$\sum_{i=i^f}^{\hat{N}_1} \hat{\tau}_1^i \geq \sum_{i=i^f}^{N_1} \tau_1^i \frac{1-\epsilon}{1+\epsilon} \geq \delta_d \frac{1-\epsilon}{1+\epsilon}. \tag{8}$$

By definition, $\hat{\rho}_1 \in \hat{\mathcal{F}}_j^{\delta}(H)$ with $\delta = \delta_d \frac{1-\epsilon}{1+\epsilon}$, and thus $\hat{\mathcal{J}}(H)$ is not $(\delta_d \frac{1-\epsilon}{1+\epsilon}, \delta_m + 2\epsilon)$-diagnosable. ∎

Lemma 2. *Given a faulty set L^f, and $\mathcal{J}(H), \hat{\mathcal{J}}(H)$ that satisfy $\vec{h}(\mathcal{P}(H), \hat{\mathcal{P}}(H)) \leq \epsilon$, if $\hat{\mathcal{J}}(H)$ is (δ_d, δ_m)-diagnosable, and $\delta_m \geq 2\epsilon$, then $\mathcal{J}(H)$ is $(\delta_d \frac{1+\epsilon}{1-\epsilon}, \delta_m - 2\epsilon)$-diagnosable.*

Proof: Directly follow from Lemma 1. ∎

Theorem 1. *Given a faulty set L^f, and $\mathcal{J}(H), \hat{\mathcal{J}}(H)$ that satisfy $h(\mathcal{P}(H), \hat{\mathcal{P}}(H)) \leq \epsilon$, if $\hat{\mathcal{J}}(H)$ is not (δ_d, δ_m)-diagnosable, then $\mathcal{J}(H)$ is not $(\delta_d \frac{1-\epsilon}{1+\epsilon}, \delta_m + 2\epsilon)$-diagnosable; if $\hat{\mathcal{J}}(H)$ is (δ_d, δ_m)-diagnosable, and $\delta_m \geq 2\epsilon$, then $\mathcal{J}(H)$ is $(\delta_d \frac{1+\epsilon}{1-\epsilon}, \delta_m - 2\epsilon)$-diagnosable.*

Proof: Directly follow from Lemma 1 and 2. ∎

E. Diagnosability Analysis Algorithm

To analyze whether H is (δ_d^*, δ_m^*)-diagnosable, the following steps can be used:

1) Randomly simulate initial states for the horizon $[0, t_{end}]$ (or $\hat{N}$) and compute the enlarged robust neighborhoods (or robust neighborhoods if possible) around them to fully cover $L^0 \times X^0$; then construct the system abstraction (see Section II-C).
2) It is guaranteed that $\vec{h}(\mathcal{P}(H), \hat{\mathcal{P}}(H)) \leq \epsilon$, where $\mathcal{P}(H) = \mathcal{P}_R(t_{end} \frac{1-\epsilon}{1+\epsilon}, H)$ and $\hat{\mathcal{P}}(H) = \hat{\mathcal{P}}_R(t_{end}, H)$ (or $\mathcal{P}(H) = \mathcal{P}_N(\hat{N} - 1, H)$ and $\hat{\mathcal{P}}(H) = \hat{\mathcal{P}}_N(\hat{N} - 1, H)$). So we can use Algorithm 1 to verify the (δ_d^*, δ_m^*)-diagnosability of H for finite horizon.
3) If $\vec{h}(\hat{\mathcal{P}}(H), \mathcal{P}(H)) \leq \epsilon$ holds as well, where $\mathcal{P}(H) = \mathcal{P}_R(t_{end} \frac{1+\epsilon}{1-\epsilon}, H)$ and $\hat{\mathcal{P}}(H) = \hat{\mathcal{P}}_R(t_{end}, H)$ (or $\mathcal{P}(H) = \mathcal{P}_N(\hat{N} - 1, H)$ and $\hat{\mathcal{P}}(H) = \hat{\mathcal{P}}_N(\hat{N} - 1, H)$), then we can use an algorithm slightly different from Algorithm 1 to falsify the (δ_d^*, δ_m^*)-diagnosability of H:
   - Line 2 is changed to: compute $\hat{\delta}_d \leftarrow \delta_d^* \frac{1+\epsilon}{1-\epsilon}, \hat{\delta}_m \leftarrow \delta_m^* - 2\epsilon$.
   - Lines 16-18 are changed to: if $\hat{\delta}_m \geq \bar{\delta}_m$, then $\hat{\mathcal{J}}(H)$ is not $(\hat{\delta}_d, \hat{\delta}_m)$-diagnosable, $\mathcal{J}(H)$ is not (δ_d^*, δ_m^*)-diagnosable.
4) It is possible that the (δ_d^*, δ_m^*)-diagnosability of $\mathcal{J}(H)$ can neither be verified nor falsified by analyzing the fault diagnosability of $\hat{\mathcal{J}}(H)$. This means the precision of the abstraction cannot provide enough information to conclude whether H is $(\hat{\delta}_d, \hat{\delta}_m)$-diagnosable. Then we can set the parameters ϵ, d_{thr} of the algorithm in [27] to smaller values, in order to construct a system abstraction with higher precision.

Algorithm 1 Given L^f and $\hat{\mathcal{J}}(H)$, verify if $\mathcal{J}(H)$ is (δ_d^*, δ_m^*)-diagnosable.

1: **procedure** VERIFICATION($\delta_d^*, \delta_m^*, \epsilon$)
2: compute $\hat{\delta}_d \leftarrow \delta_d^* \frac{1-\epsilon}{1+\epsilon}, \hat{\delta}_m \leftarrow \delta_m^* + 2\epsilon$
3: **for** $j \leftarrow 1$ to M **do**
4: compute $\hat{\mathcal{F}}_j(H)$, the set of F_j-faulty trajectories
5: compute $\hat{\mathcal{F}}_j^{\hat{\delta}_d}(H)$, the set of $F_j\hat{\delta}_d$-faulty trajectories
6: **end for**
7: compute $\hat{\mathcal{F}}_0(H)$, the set of normal trajectories
8: **for** $j \leftarrow 1$ to M **do**
9: **for** $i \leftarrow 0$ to M **do**
10: **if** $i \neq j$ **then**
11: compute $\bar{\delta}_m^{i,j} \leftarrow \inf_{\hat{\rho}_1 \in \hat{\mathcal{F}}_j^{\hat{\delta}_d}(H)} \inf_{\hat{\rho}_2 \in \hat{\mathcal{F}}_i(H)} d_{\Sigma^*}(\hat{\pi}_1, \hat{\pi}_2)$
▷ $\hat{\pi}_1, \hat{\pi}_2$ are the projected label sequences corresponding to $\hat{\rho}_1, \hat{\rho}_2$.
12: **end if**
13: **end for**
14: **end for**
15: compute $\bar{\delta}_m \leftarrow \inf_{1 \leq j \leq M} \inf_{0 \leq i \leq M} \bar{\delta}_m^{i,j}$
16: **if** $\hat{\delta}_m < \bar{\delta}_m$ **then**
17: $\hat{\mathcal{J}}(H)$ is $(\hat{\delta}_d, \hat{\delta}_m)$-diagnosable, $\mathcal{J}(H)$ is (δ_d^*, δ_m^*)-diagnosable
18: **end if**
19: **end procedure**

F. Diagnosers

The system abstraction constructed in Section II-C allows us to build a diagnoser for the system. The proposed approach to system abstraction and diagnoser construction is trajectory-based, which is disparate from other approaches in the literature. For instance, the work [29] proposed to construct a durational graph as an abstraction of the original system in order to analyze the diagnosability, which is fast but conservative, and the constructed durational graph requires further analysis of the timed language. In contrast, the trajectory-based abstraction in the present work can be arbitrarily precise, and the constructed diagnoser works in the following transparent way: It stores a finite list of candidate trajectories whose location sequences can be reached by H, and keeps narrowing down the list by observing the timed event sequences of H till a decision is made.

We construct the diagnoser as a hybrid automaton $H_d = (L_d \times X_d, L_d^0 \times X_d^0, D_d, E_d, Inv_d)$:

- Let $\{\hat{\rho}_k\}_{k=1}^K \subset \hat{\mathcal{J}}(H)$ denote the set of (virtual) trajectories that extend to the end of the time horizon, where $\hat{\rho}_k = \{(\hat{e}_k^i, \hat{\ell}_k^i, \hat{x}_k^i, \hat{\tau}_k^i)\}_{i=0}^{\hat{N}_k}$. The state space of the diagnoser is defined as $L_d := 2^{\{1,\ldots,K\}} \times \{0, 1, 2, \ldots\}$, $X_d := \mathbb{R}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

Example. *A location $\ell_d \in L_d$ can be $(\{1,6\},2)$. The diagnoser being at ℓ_d means that the location sequence reached by the running trajectory of H matches that of $\hat{\rho}_1$ or $\hat{\rho}_6$, and currently 2 observable events have been triggered after the starting signal.*

- Let $\hat{s}_k = \{(\hat{\Delta}_k^i, \hat{\psi}_k^i)\}_{i=0}^{\hat{N}_k}$ be the label sequence produced by $\hat{\rho}_k$, which possesses the sequence of starting signal and observable output symbols $\{\hat{\psi}_k^{i_{k,n}}\}_{n=0}^{N_k} \subset \Psi_o \cup \{\iota\}, N_k \leq \hat{N}_k$. Then the projected label sequence $\Pi(\hat{s}_k)$ is $\hat{\pi}_k = \{(\Delta_k^n, \psi_k^n)\}_{n=0}^{N_k}$, where $\psi_k^n = \hat{\psi}_k^{i_{k,n}}$, $\Delta_k^0 = 0$, and $\Delta_k^n = \Sigma_{i=i_{k,n-1}+1}^{i_{k,n}} \hat{\Delta}_k^i$ for all $n \geq 1$. For clarity, we denote the index $i_{k,n}$ as $ind_k(n)$.
- Define the fault labels $W := 2^{\mathbb{F}}$, where $\mathbb{F} := \{F_1, \ldots, F_M\}$ are the modeled M types of faults. Each pair $(k,n), n \in \{0,1,\ldots,N_k\}$ possesses a fault label $w_{(k,n)} \in W$ as the collection of all faults made by the sequence $\{\hat{\rho}_k^i\}_{i=0}^{ind_k(n)}$. Given a location $\ell_d = ([k],n) \in 2^{\{1,\ldots,K\}} \times \{0,1,2,\ldots\}$ such that $n \leq N_k$ for all $k \in [k]$, the set $\{w_{(k,n)} | k \in [k]\}$ is referred to as $[w]_{\ell_d}$.

  Example. *When $\ell_d = (\{1,6\},2)$, $[w]_{\ell_d}$ can be $\{\{\},\{F_1\}\}$, meaning that as far as the second observable event is triggered after the starting signal, $\hat{\rho}_1$ is normal while $\hat{\rho}_6$ makes the fault F_1.*
- $L_d^0 \times X_d^0 := (\{1,\ldots,K\},0) \times \{0\}$.
- $D_d : \dot{x} = 1$ for all the locations $\ell_d \in L_d$.
- Given a location $\ell_d = ([k],n)$, the observable output symbols $\{\psi_k^{n+1} | k \in [k], N_k \geq n+1\}$ (recall that $\hat{\pi}_k = \{(\Delta_k^n, \psi_k^n)\}_{n=0}^{N_k}$ is the projected label sequence produced by $\hat{\rho}_k$) can be classified into q distinct symbols $\{\psi_{(1)}, \ldots, \psi_{(q)}\}$. Assume $\{\psi_k^{n+1} | k \in [k], N_k \geq n+1\}$ is not empty, i.e., $q \geq 1$, then we model events for each $\psi \in \{\psi_{(1)}, \ldots, \psi_{(q)}\}$ as follows:
  Consider the observable output symbol $\psi_{(1)}$. Define the set of accumulated dwell time

$$[\Delta]_1 := \{\Delta_k^{n+1} | k \in [k], N_k \geq n+1, \psi_k^{n+1} = \psi_{(1)}\},$$

  and $I_1 := \bigcup_{\Delta \in [\Delta]_1} B(\Delta, \frac{1}{2}\hat{\delta}_m)$, where $\hat{\delta}_m$ is the measurement uncertainty under which $\hat{\mathcal{J}}(H)$ is diagnosable, $B(\Delta, \frac{1}{2}\hat{\delta}_m)$ is the relative time metric ball cantered at Δ with the radius $\frac{1}{2}\hat{\delta}_m$. Let z_1 be the number of subsets of $[\Delta]_1$. For all $[\Delta]_{1,j} \in 2^{[\Delta]_1}, j \leq z_1$, define

$$I_{1,j} := \bigcap_{\Delta \in [\Delta]_{1,j}} B(\Delta, \frac{1}{2}\hat{\delta}_m) \setminus \bigcup_{\Delta \in [\Delta]_1 \setminus [\Delta]_{1,j}} B(\Delta, \frac{1}{2}\hat{\delta}_m).$$

  It can be easily proved that $\{I_{1,j}\}_{j=1}^{z_1}$ are disjoint, and $\bigcup_{j=1}^{z_1} I_{1,j} = I_1$. Also, by definition the following properties hold:

  1) for all $\bar{\Delta} \in I_{1,j}, \Delta \in [\Delta]_{1,j}$, it is satisfied $\bar{\Delta} \in B(\Delta, \frac{1}{2}\hat{\delta}_m)$;
  2) for all $\bar{\Delta} \in I_{1,j}, \Delta \in [\Delta]_1 \setminus [\Delta]_{1,j}$, it is satisfied $\bar{\Delta} \notin B(\Delta, \frac{1}{2}\hat{\delta}_m)$.

  Example. *Given $\ell_d = (\{1,6\},2)$, if $\psi_1^3 = \psi_6^3 = \psi_{(1)}$, then $[\Delta]_1 = \{\Delta_1^3, \Delta_6^3\}$. Assume $[\Delta]_1 = \{50, 70\}$. Then $[\Delta]_{1,1} = \{50,70\}$, $[\Delta]_{1,2} = \{50\}$, $[\Delta]_{1,3} = \{70\}$; and $I_1 = B(50, \frac{1}{2}\hat{\delta}_m) \cup B(70, \frac{1}{2}\hat{\delta}_m)$, which can be partitioned to $I_{1,1} = B(50, \frac{1}{2}\hat{\delta}_m) \cap B(70, \frac{1}{2}\hat{\delta}_m)$, $I_{1,2} = B(50, \frac{1}{2}\hat{\delta}_m) \setminus B(70, \frac{1}{2}\hat{\delta}_m) = B(50, \frac{1}{2}\hat{\delta}_m) \setminus I_{1,1}$ and $I_{1,3} = B(70, \frac{1}{2}\hat{\delta}_m) \setminus B(50, \frac{1}{2}\hat{\delta}_m) = B(70, \frac{1}{2}\hat{\delta}_m) \setminus I_{1,1}$.*

  For convenience, we also define

$$\begin{aligned} [k']_{1,j} &:= \{k \in [k] | N_k \geq n+1, \\ &\quad \psi_k^{n+1} = \psi_{(1)}, \Delta_k^{n+1} \in [\Delta]_{1,j}\}, \\ \ell'_{d1,j} &:= ([k']_{1,j}, n+1). \end{aligned}$$

  Then we can model events $e_{d1,j} = (\ell_d, \ell'_{d1,j}, I_{1,j}, r)$ for all $j \leq z_1$ and $I_{1,j}$ not empty, where $r(x) = 0$ for any $x \in \mathbb{R}$.
  By using the same way, events in E_d can be modeled for other $\psi \in \{\psi_{(1)}, \ldots, \psi_{(q)}\}$, and for other locations $\ell_d \in L_d$.
- $Inv_d(\ell_d) := \mathbb{R}$ for all $\ell_d \in L_d$.
- If and only if H_d reaches a location ℓ_d such that $w_{(k,n)} = \{F_j\}$ for all $w_{(k,n)} \in [w]_{\ell_d}$, the diagnoser raises an alarm for the fault F_j, which means F_j is detected and isolated.

The diagnoser H_d starts operating at the same time as H. The continuous system state x_d is the time. The discrete system state ℓ_d updates according to the triggered events $e_d \in E_d$. An event e_d is triggered if and only if an output symbol ψ with accumulated dwell time $\bar{\Delta}$ (since the preceding observed output symbol) is observed from H, such that ψ is identical to the symbol that models e_d, and also $\bar{\Delta}$ satisfies the guard condition of e_d.

We assume the abstraction $\hat{\mathcal{J}}(H)$ is $(\hat{\delta}_d, \hat{\delta}_m)$-diagnosable, $\vec{h}(\mathcal{P}(H), \hat{\mathcal{P}}(H)) \leq \epsilon$, and $\hat{\delta}_m \geq 2\epsilon$. By Theorem 1, H is $(\hat{\delta}_d \frac{1+\epsilon}{1-\epsilon}, \hat{\delta}_m - 2\epsilon)$-diagnosable. We thus assume that the measurement uncertainty of time intervals for H is at most $(\hat{\delta}_m - 2\epsilon)$. Specifically, any measured time interval must lie inside a ball of radius $(\frac{1}{2}\hat{\delta}_m - \epsilon)$ centered at the true value.

Proposition 6. *If the trajectory of H, $\rho \in \mathcal{J}(H)$, is $F_j\delta_d^*$-faulty, where $\delta_d^* = \hat{\delta}_d \frac{1+\epsilon}{1-\epsilon}$, then the F_j fault alarm is already raised by the diagnoser H_d.*

Proof: For any $\rho \in \mathcal{J}(H)$, there exists $\hat{\rho} \in \hat{\mathcal{J}}(H)$ such that the produced timed event sequences satisfy $d_\chi(p, \hat{p}) \leq \epsilon$. Hence, ρ is $F_j\delta_d^*$-faulty implies that $\hat{\rho}$ is $F_j\hat{\delta}_d$-faulty. By Prop. 5, we have $d_{\Sigma^*}(\pi, \hat{\pi}) \leq \epsilon$, where $\pi, \hat{\pi}$ are the projected label sequences produced by $\rho, \hat{\rho}$. Let $\bar{\pi}$ be the measurement of π. Then $d_{\Sigma^*}(\bar{\pi}, \hat{\pi}) \leq d_{\Sigma^*}(\pi, \hat{\pi}) + d_{\Sigma^*}(\bar{\pi}, \pi) \leq \frac{1}{2}\hat{\delta}_m$.

Since $\hat{\mathcal{J}}(H)$ is $(\hat{\delta}_d, \hat{\delta}_m)$-diagnosable, for any $\hat{\rho}'$ that is normal or F_i-faulty ($F_i \neq F_j$), the projected label sequences produced by $\hat{\rho}, \hat{\rho}'$ must satisfy $d_\Sigma(\hat{\pi}, \hat{\pi}') > \hat{\delta}_m$. Thus, $d_\Sigma(\bar{\pi}, \hat{\pi}') > \frac{1}{2}\hat{\delta}_m$. Namely, the measurement of the projected label sequence of ρ must be at least $\frac{1}{2}\hat{\delta}_m$ away from the projected label sequence of any normal or F_i-faulty trajectory.

Suppose $w_{(k,n)} \neq \{F_j\}$ for some $(k,n) \in \ell_d$, where $\ell_d = ([k],n)$ is the last discrete state of the diagnoser updated according to $\bar{\pi}$. By the construction of H_d, there must exist $\hat{\rho}'$ as a sub-trajectory of $\hat{\rho}_k$, such that $\hat{\rho}'$ is normal or F_i-faulty ($F_i \neq F_j$), and its projected label sequence satisfies $d_{\Sigma^*}(\bar{\pi}, \hat{\pi}') \leq \frac{1}{2}\hat{\delta}_m$.

By contradiction, the fault alarm for F_j should be raised no later than the last discrete state reached by the diagnoser according to $\bar{\pi}$. ■

Proposition 7. *If the F_j fault alarm is raised by the diagnoser for the trajectory $\rho \in \mathcal{J}(H)$, then ρ must be F_j-faulty.*

Proof: For any $\rho \in \mathcal{J}(H)$, there exists $\hat{\rho} \in \hat{\mathcal{J}}(H)$ such that the produced timed event sequences satisfy $d_\chi(p, \hat{p}) \leq \epsilon$. Let $\hat{\rho}_k \in \hat{\mathcal{J}}(H)$ be a (virtual) trajectory extending to the end of the time horizon such that $\hat{\rho}$ is a sub-trajectory of $\hat{\rho}_k$. The label sequence produced by $\hat{\rho}_k$ is $\hat{s}_k = \{(\hat{\Delta}_k^i, \hat{\psi}_k^i)\}_{i=0}^{\hat{N}_k}$. The diagnoser operates according to the measurement $\bar{\pi}$ of the projected label sequence π produced by ρ. By the construction of H_d, clearly $(k,0) \in \ell_d^0$, where ℓ_d^i ($i \geq 0$) denotes the $(i+1)^{th}$ location reached by H_d during its operation. Let $\hat{e}_k^{i_1}$ be the first observable event in E triggered by $\hat{\rho}_k$ after the starting signal, whose associated output symbol is $\hat{\psi}_k^{i_1} \in \Psi_o$, and the accumulated dwell time since the initialization is $\Delta := \sum_{i=1}^{i_1} \hat{\Delta}_k^i$.

Write the projected label sequence $\hat{\pi}$ produced by $\hat{\rho}$ as a sequence $(0,\iota), (\Delta, \psi), \cdots$, and the measurement $\bar{\pi}$ of π as $(0,\iota), (\bar{\Delta}, \psi), \cdots$, then $\psi = \hat{\psi}_k^{i_1}$, and $\bar{\Delta}$ is on the guard of the first event triggered by the diagnoser. We write this event as $e_{d1,j} = (\ell_d, \ell'_{d1,j}, I_{1,j}, r)$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

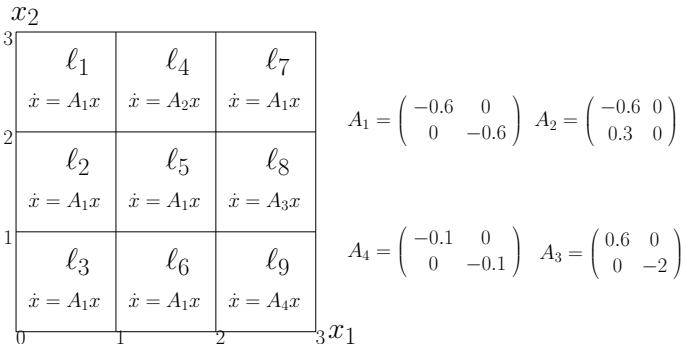


Fig. 2. The invariant sets and dynamics of the hybrid automaton.

like in the construction of E_d. Accordingly, $\hat{\psi}_k^{i_1}$ corresponds to $\psi_{(1)}$ in the construction of $e_{d1,j}$, and $\ell_d = \ell_d^0$, $\ell'_{d1,j} = \ell_d^1$, $\bar{\Delta} \in I_{1,j}$. According to the second property of guards in the construction of $e_{d1,j}$, if $\Delta \in [\Delta]_1 \setminus [\Delta]_{1,j}$, then $\bar{\Delta} \notin B(\Delta, \frac{1}{2}\hat{\delta}_m)$. We already have $\Delta = \sum_{i=1}^{i_1} \hat{\Delta}_k^i \in [\Delta_1]$ by definition. Moreover, it follows from $d_{\Sigma^*}(\bar{\pi}, \hat{\pi}) \leq \frac{1}{2}\hat{\delta}_m$ (see the proof of Prop. 6) that $\bar{\Delta} \in B(\Delta, \frac{1}{2}\hat{\delta}_m)$. Therefore, $\Delta \in [\Delta]_{1,j}$, $(k,1) \in \ell'_{d1,j} = \ell_d^1$. With similar argument, it can be proved that for any (Δ, ψ) output by the n^{th} observable event of $\hat{\rho}_k$ triggered after the starting signal, the updated location of the diagnoser must contain (k,n). If H_d reaches a location whose fault labels are all $\{F_j\}$, which implies $w_{(k,n)} = \{F_j\}$, then clearly $\hat{\rho}$ is F_j-faulty, and thus ρ is F_j-faulty. ■

G. Numerical Example of Diagnosability Verification

Consider a 2-dimensional hybrid automaton H with 9 locations. The invariant sets and dynamics are visualized in Figure 2. Guards are boundaries of the invariant sets. Reset map for continuous state is the identity matrix. Some trajectories initiated from the set $\{(x_1, x_2) | 2.2 \leq x_1 \leq 2.3, 2.2 \leq x_2 \leq 2.3\}$ are simulated for the time horizon $[0, 1.5]$ as shown in Figure 3. All the events in E has the unified output symbol α. The location ℓ_8 is designated as faulty. The initial set leads to both normal and faulty trajectories.

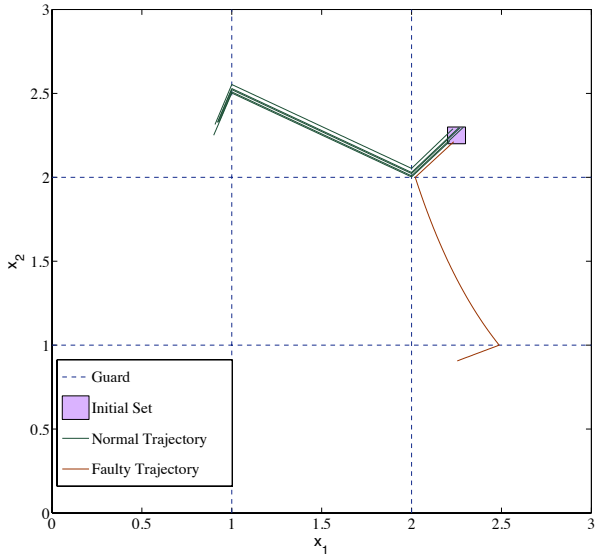


Fig. 3. Normal or faulty trajectories of the hybrid automaton.

We construct the system abstraction as in Section II-C by covering the initial set with the enlarged robust neighborhoods computed for 11 simulated initial states, where the parameter ϵ in relative time metric is set to 0.1 (see Figure 4).

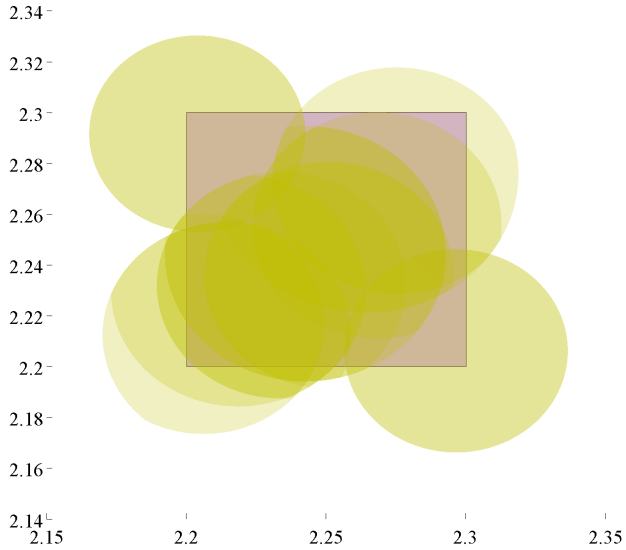


Fig. 4. The initial set is covered by enlarged robust neighborhoods.

Since $\hat{\mathcal{J}}(H)$ contains a finite number of simulated trajectories, it is easy to verify that 5 trajectories are normal (reaching ℓ_7, ℓ_4, ℓ_1), while 6 trajectories are faulty (reaching ℓ_7, ℓ_8, ℓ_9).

There are two simulated trajectories

$$\begin{aligned} \rho_1 &= (e^0, 7, [2.2062, 2.2166]', 0.1635), \\ &\quad (e^1, 4, [2, 2.0095]', 1.1552), (e^2, 1, [1, 2.5095]', 0.1813), \\ \rho_2 &= (e^0, 7, [2, 2964, 2.2062]', 0.1635), \\ &\quad (e^1, 8, [2.0818, 2]', 0.3466), (e^2, 9, [2.5630, 1]', 0.9899), \end{aligned}$$

respectively corresponding to the projected label sequences $\pi_1 = (0, \iota), (0.1635, \alpha), (1.1552, \alpha)$ and $\pi_2 = (0, \iota), (0.1635, \alpha), (0.3466, \alpha)$. Hence, although ρ_2 goes faulty to ℓ_8, it is not possible to tell it immediately apart from the normal ρ_1 by monitoring only the first two output symbols and the associated time intervals. Instead, we need to wait longer. With Algorithm 1 it can be verified that $\hat{\mathcal{J}}(H)$ is $(0.3466, 0.5384)$-diagnosable, and $\mathcal{J}(H)$ is $(0.4236, 0.3384)$-diagnosable. Therefore, by observing the output symbols and measuring the time intervals in between with the measurement uncertainty 0.3384 in relative time metric, any trajectory of H that enters ℓ_8 as the second location at the time instant t^f can be diagnosed before the time window $[t^f, t^f + 0.4236]$ runs out.

IV. Conclusion

In this paper, we propose a diagnosability notion for hybrid automata that conveys the maximum delay in detecting and isolating faults under given measurement uncertainty in time intervals. We prove a quantitative relation on such (δ_d, δ_m)-diagnosability of a hybrid automaton and its system abstraction, whose timed language approximately includes or is approximately equivalent to that of the original system. By analyzing the (δ_d, δ_m)-diagnosability of the system abstraction especially when it has only finitely many trajectories extending to the time horizon of interest, diagnosability analysis and diagnoser construction of the original system can be

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

simplified to a great extent. We present a method to construct such system abstractions by using the robust test generation and coverage idea, which can be implemented automatically with the Matlab toolbox STRONG [30] for linear systems. An example is illustrated, reducing the finite-horizon diagnosability analysis of a system with infinitely many trajectories to finite.

LIST OF SYMBOLS

H	hybrid automaton
$L \times X$	state space
D	dynamics
$L^0 \times X^0$	initial set
E	events
Inv	invariant sets
ℓ	location (discrete state)
x	continuous state
e	event
r	reset map
g	guard
G_ℓ	guards (in $\ell \in L$)
$\xi_\ell(t, x_\ell^0)$	dynamical system solution
ρ	trajectory
J	a set of trajectories
p	timed event sequence
P	timed language
Δ	dwell time
ψ	output symbol
$\mathbb{R}$	real numbers
Ψ_o	observable output symbols
$\emptyset$	unobservable output symbol
ι	starting signal
Σ	labels
s	label sequence
$\mathcal{J}$	trajectories of interest
$\mathcal{P}$	timed language produced by $\mathcal{J}$
Π	projection map
π	projected label sequence
L^f	faulty set
F_j	a type of fault
δ_d	delay parameter
δ_m	measurement uncertainty

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

REFERENCES

[1] J. J. Gertler, "Survey of model-based failure detection and isolation in complex plants," *Control Systems Magazine, IEEE*, vol. 8, no. 6, pp. 3–11, 1988.

[2] M. O. Cordier, P. Dague, F. Levy, J. Montmain, M. Staroswiecki, and L. Trave-Massuyes, "Conflicts versus analytical redundancy relations a comparative analysis of the model based diagnosis approach from the artificial intelligence and automatic control perspectives," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 34, no. 5, pp. 2163–2177, 2004.

[3] M. A. Massoumnia, G. C. Verghese, and A. S. Willsky, "Failure detection and identification," *Automatic Control, IEEE Transactions on*, vol. 34, no. 3, pp. 316–321, March 1989.

[4] E. Chow and A. S. Willsky, "Analytical redundancy and the design of robust failure detection systems," *Automatic Control, IEEE Transactions on*, vol. 29, no. 7, pp. 603–614, 1984.

[5] J. Chen and R. J. Patton, *Robust model-based fault diagnosis for dynamic systems*. Kluwer Academic Publishers, 1999.

[6] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy—a survey and some new results," *Automatica*, vol. 26, no. 3, pp. 459–474, May 1990.

[7] R. Isermann and P. Ballé, "Trends in the application of model-based fault detection and diagnosis of technical processes," *Control Engineering Practice*, vol. 5, no. 5, pp. 709 – 719, 1997.

[8] M. Bayoudh, L. Travé-Massuyes, and X. Olive, "Hybrid systems diagnosability by abstracting faulty continuous dynamics," in *Proceedings of the 17th International Principles of Diagnosis Workshop*, 2006, pp. 9–15.

[9] M. Bayoudh, L. Travé-Massuyes, X. Olive, and T. A. Space, "Hybrid systems diagnosis by coupling continuous and discrete event techniques," in *Proceedings of the IFAC World Congress, Seoul, Korea*, 2008, pp. 7265–7270.

[10] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete-event systems," *Automatic Control, IEEE Transactions on*, vol. 40, no. 9, pp. 1555–1575, 1995.

[11] M. D. Di Benedetto, S. Di Gennaro, and A. D'Innocenzo, "Diagnosability verification for hybrid automata and durational graphs," in *Decision and Control, 2007 46th IEEE Conference on*, 2007, pp. 1789–1794.

[12] S. Tripakis, "Fault diagnosis for timed automata," in *Formal Techniques in Real-Time and Fault-Tolerant Systems*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2002, vol. 2469, pp. 205–221.

[13] M. D. Di Benedetto, S. Di Gennaro, and A. D'Innocenzo, "Diagnosability of hybrid automata with measurement uncertainty," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, 2008, pp. 1042–1047.

[14] A. Abate, A. D'Innocenzo, and M. D. Di Benedetto, "Approximate abstractions of stochastic hybrid systems," *Automatic Control, IEEE Transactions on*, vol. 56, no. 11, pp. 2688–2694, 2011.

[15] A. D'Innocenzo, A. A. Julius, M. D. Di Benedetto, and G. J. Pappas, "Approximate timed abstractions of hybrid automata," in *Decision and Control, 2007 46th IEEE Conference on*, 2007, pp. 4045–4050.

[16] A. D'Innocenzo, A. A. Julius, G. J. Pappas, M. D. Di Benedetto, and S. Di Gennaro, "Verification of temporal properties on hybrid automata by simulation relations," in *Decision and Control, 2007 46th IEEE Conference on*, 2007, pp. 4039–4044.

[17] G. E. Fainekos, A. Girard, and G. J. Pappas, "Temporal logic verification using simulation," in *Formal Modeling and Analysis of Timed Systems*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2006, vol. 4202, pp. 171–186.

[18] A. Girard, "Approximately bisimilar finite abstractions of stable linear systems," in *in Hybrid Systems: Computation and Control, ser. Lecture*, 2007.

[19] A. Girard, G. Pola, and P. Tabuada, "Approximately bisimilar symbolic models for incrementally stable switched systems," *Automatic Control, IEEE Transactions on*, vol. 55, no. 1, pp. 116–126, 2010.

[20] G. Pola, A. Girard, and P. Tabuada, "Approximately bisimilar symbolic models for nonlinear control systems," *Automatica*, vol. 44, no. 10, pp. 2508 – 2516, 2008.

[21] G. Reissig, "Computing abstractions of nonlinear systems," *Automatic Control, IEEE Transactions on*, vol. 56, no. 11, pp. 2583–2598, Nov 2011.

[22] P. Tabuada, "An approximate simulation approach to symbolic control," *Automatic Control, IEEE Transactions on*, vol. 53, no. 6, pp. 1406–1418, 2008.

[23] M. Zamani, G. Pola, M. Mazo, and P. Tabuada, "Symbolic models for nonlinear control systems without stability assumptions," *Automatic Control, IEEE Transactions on*, vol. 57, no. 7, pp. 1804–1809, July 2012.

[24] R. Alur, C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P. H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine, "The algorithmic analysis of hybrid systems," *Theoretical Computer Science*, vol. 138, pp. 3–34, 1995.

[25] M. Heymann, F. Lin, G. Meyer, and S. Resmerita, "Analysis of zeno behaviors in hybrid systems," in *In: Proceedings of the 41st IEEE Conference on Decision and Control, Las Vagas, NV (2002)*, 2002, pp. 2379–2384.

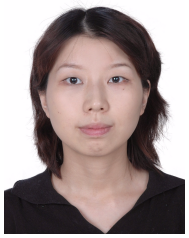
[26] A. A. Julius, G. E. Fainekos, M. Anand, I. Lee, and G. J. Pappas, "Robust test generation and coverage for hybrid systems," in *In Proc. of the 10th International Workshop on Hybrid Systems: Computation and Control*. Springer, 2007, pp. 329–342.

[27] Y. Deng and A. A. Julius, "Safe neighborhood computation for hybrid system verification," in Proceedings 4th Workshop on Hybrid Autonomous Systems, Grenoble, France, 12-13 April 2014, ser. Electronic Proceedings in Theoretical Computer Science, vol. 174, 2015, pp. 1–12.

[28] M. O. Cordier, L. Travé-Massuyès, and X. Pucel, "Comparing diagnosability in continuous and discrete-event systems," in *Proceedings of the 17th International Workshop on Principles of Diagnosis (DX-06)*, 2006, pp. 55–60.

[29] M. D. Di Benedetto, S. Di Gennaro, and A. D'Innocenzo, "Verification of hybrid automata diagnosability by abstraction," *Automatic Control, IEEE Transactions on*, vol. 56, no. 9, pp. 2050–2061, Sept 2011.

[30] Y. Deng, A. Rajhans, and A. A. Julius, "Strong: A trajectory-based verification toolbox for hybrid systems," in *Quantitative Evaluation of Systems*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, vol. 8054, pp. 165–168.



Yi Deng attended Rensselaer Polytechnic Institute, Troy, NY in 2010, where she is currently pursuing the Ph.D. degree in the Department of Electrical, Computer, and Systems Engineering. Her research interests include formal verification of reachability and safety properties, and fault diagnosis for hybrid systems.

Alessandro D'Innocenzo is Assistant Professor in the Department of Information Engineering, Computer Science and Mathematics at the University of L'Aquila. In 2005 he was recipient of Fondazione Filauro award for PhD students. In 2007 he obtained the PhD degree in Electrical and Information Engineering from University of L'Aquila, and accomplished the International Curriculum Option of Doctoral Studies in Hybrid Control for Complex, Distributed and Heterogeneous Embedded Systems. In 2007 and 2009 he has been Postdoctoral Researcher in the Department of Electrical and Information Engineering of University of L'Aquila. In 2008 he has been Postdoctoral Researcher in the Department of Electrical and Systems Engineering of University of Pennsylvania. His research focuses on control theory and in particular, hybrid systems, formal verification and networked control, with applications to air traffic management, building automation and communication systems.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2455111, IEEE Transactions on Automatic Control

Maria Domenica Di Benedetto obtained the "Dr. Ing." degree (summa cum laude) of Electrical Engineering and Computer Science, University of Roma "La Sapienza", in 1976. In 1981, she obtained the degree "Docteur-Ingenieur" and in 1987 the degree "Doctorat d'Etat Sciences", Universit de Paris-Sud (Orsay, France). Since 1994, she has been Professor of Control Theory at University of l'Aquila. Since 2002, she has been IEEE Fellow. She is the PI and Director of the Center of Excellence for Research DEWS "Architectures and Design methodologies for Embedded controllers, Wireless interconnect and System-on-Chip". Since 1995, she is Member of the Scientific Committee of the Center of Excellence for Research CETEMPS, University of L'Aquila. She is co-founder and member of the Governing Board of WEST Aquila S.r.L.. She is Member of the Board of Fondazione MIRROR since 2008. Her research interests revolve around nonlinear control and hybrid systems, with applications to automotive and air traffic control.

Stefano Di Gennaro obtained the degree in Nuclear Engineering in 1987 (summa cum laude), and the Ph.D. degree in System Engineering in 1992, both from the University of Rome "La Sapienza", Rome, Italy. In October 1990 he joined the Department of Electrical Engineering, University of L'Aquila, as Assistant Professor of Automatic Control. Since 2001, he has been Associate Professor of Automatic Control at the University of L'Aquila. He holds courses on Automatic Control and Nonlinear Control. In 1986 he was visiting scientist at the Nuclear Research Center ENEA – Casaccia. He has been visiting professor at the Laboratoire des Signaux et Systemes, CNRS–Paris, of the Department of Electrical Engineering of the Princeton University and of the Department of Electrical Engineering and Computer Science of the University of California at Berkeley, USA, and of the Centro de Investigacion y Estudios Avanzados del IPN, Unidad Ciudad de Mexico and Unidad Guadalajara, Mexico. He is working in the area of hybrid systems, regulation theory, and applications of nonlinear control.

A. Agung Julius received the Ph.D. degree in applied mathematics from the University of Twente, The Netherlands, in 2005.

He joined the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, as an Assistant Professor in December 2008. From 2005 to 2008, he was a Postdoctoral Researcher at the University of Pennsylvania. His research interests include systems and control, systems biology, stochastic models in systems biology, control of biological systems, hybrid systems, and mathematical systems theory.

Dr. Julius received an NSF CAREER award in 2010.