2013 IEEE Conference on Computer Aided Control System Design (CACSD)
Part of 2013 IEEE Multi-Conference on Systems and Control
Hyderabad, India, August 28-30, 2013

Online Failure Diagnosis of Stochastic Discrete Event Systems

Jun Chen, *Student Member, IEEE* and Ratnesh Kumar, *Fellow, IEEE*

Abstract— **This paper deals with the detection of (permanent) fault in the setting of stochastic discrete-event systems (DESs) under partial observability of events. Prior works have only studied the verification of the stochastic diagnosability (S-Diagnosability) property. To the best of our knowledge, this is a first paper that investigates the online detection schemes and also introduces the notions of their missed detections (MDs) and false alarms (FAs), and we establish that S-Diagnosability is a necessary and sufficient condition for achieving any desired levels of MD and FA rates. Next we provide a detection scheme, that can achieve the specified MD and FA rates, based on comparing a suitable detection statistic, that we define, with a suitable detection threshold, that we algorithmically compute. We also algorithmically compute the corresponding detection delay bound. The detection scheme also works for non-S-Diagnosable systems, with the difference that in this case only any FA rate can be met, and there exists a minimum MD rate that increases as FA rate is decreased.**

I. Introduction

The problem of failure detection in discrete-event systems (DESs) has been widely studied [1]-[9]. The notion of stochastic diagnosability, *S-Diagnosability*, was proposed in [5] (where it is referred as AA-diagnosability). A necessary and sufficient test for checking S-Diagnosability that has a polynomial complexity was presented in [1]. These prior works have only studied the verification of the S-Diagnosability property; a technique online fault detection hasn't yet been examined in literature. To the best of our knowledge, this is a first paper that investigates the online detection schemes for stochastic DESs and also introduces the notions of their missed detections (MDs) and false alarms (FAs). Due to the probabilistic nature of the problem, MDs and FAs are possible even for S-Diagnosable systems, and we establish that S-Diagnosability is a necessary and sufficient condition for achieving any desired levels of MD and FA rates.

Next we present a detection scheme, that can achieve the specified MD and FA rates, based on comparing a suitable detection statistic with a suitable detection threshold that we algorithmically compute. We also algorithmically compute the corresponding detection delay bound. The idea is that given any observation (of partially observed events), the detector recursively computes the conditional probability of the nonoccurrence of a fault and issues a "fault" decision if the probability of the nonoccurrence of a fault falls below an appropriately chosen threshold, and issues "no-decision" otherwise. We show that the existence of a detector for any desired MD and FA rates is a necessary and sufficient condition for the system to be S-Diagnosable. Algorithms for determining the detection scheme parameters of detection threshold and detection delay bound for the specified MD and FA rates requirement are also presented, based on the construction of an extended observer. Our detection strategy works for S-Diagnosable system as well as non-S-Diagnosable systems in the same manner. For a non-S-Diagnosable system an arbitrary performance requirement is achievable only for the FA rate, whereas a lower bound exists for the achievable MD rate that is a function of the FA rate, and increases as FA rate is decreased. A variant of the above mentioned algorithm is also presented to compute an upper bound for the minimum achievable MD rate for a non-S-Diagnosable system.

The rest of this paper is organized as following. The notations and some preliminaries are presented in Section II, followed by the proposed online fault detector and algorithms in Section III. Section IV concludes the paper.

II. Notations and Preliminaries

A. Stochastic DESs

For an event set Σ, define $\overline{\Sigma} := \Sigma \cup \{\epsilon\}$, where ϵ represents "no-event", and Σ^* denotes the set of all finite length event sequences over Σ, including ϵ. A member of Σ^* is called a *trace*. Denote as $s \leq t$ if $s \in \Sigma^*$ is a prefix of $t \in \Sigma^*$, and use $|s|$ to denote the number of events in s or the length of s. A subset of Σ^* is called *language*. For $L \subseteq \Sigma^*$, its prefix-closure, denoted as $pr(L)$, is defined as $pr(L) := \{s \in \Sigma^* | \exists t \in L : s \leq t\}$. L is said to be prefix-closed (or simply closed) if $pr(L) = L$, i.e., whenever L contains a trace, it also contains all the prefixes of that trace. For $s \in \Sigma^*$ and $L \subseteq \Sigma^*$, $L \backslash s$ denotes the set of traces in L after s and is defined as $L \backslash s := \{t \in \Sigma^* | st \in L\}$.

A stochastic DES can be modeled as a *stochastic automaton* G which is denoted by $G = (X, \Sigma, \alpha, x_0)$, where X is the set of states, Σ is the finite set of events, $x_0 \in X$ is the initial state, and $\alpha : X \times \Sigma \times X \rightarrow [0, 1]$ is the transition probability function [10], satisfying: $\forall x \in X, \sum_{\sigma \in \Sigma} \sum_{x' \in X} \alpha(x, \sigma, x') = 1$. G is said to be non-stochastic if $\alpha : X \times \Sigma \times X \rightarrow \{0, 1\}$, and a non-stochastic DES is said to be deterministic if $\forall x \in X, \sigma \in \Sigma, \sum_{x' \in X} \alpha(x, \sigma, x') \leq 1$. The transition probability function α can be extended from domain $X \times \Sigma \times X$ to $X \times \Sigma^* \times X$ recursively as follows: $\forall x_i, x_j \in X, s \in \Sigma^*, \sigma \in \Sigma, \alpha(x_i, s\sigma, x_j) = \sum_{x_k \in X} \alpha(x_i, s, x_k)\alpha(x_k, \sigma, x_j)$, and $\alpha(x_i, \epsilon, x_j) = 1$ if $x_i = x_j$ and 0 otherwise. Define a *transition* in G as a triple $(x_i, \sigma, x_j) \in X \times \Sigma \times X$ where

The research was supported in part by the National Science Foundation under the grants NSF-CCF-0811541, and NSF-ECCS-0926029.

The authors are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: junchen@iastate.edu; rkumar@iastate.edu).

978-1-4799-1565-1/13/$31.00 ©2013 IEEE

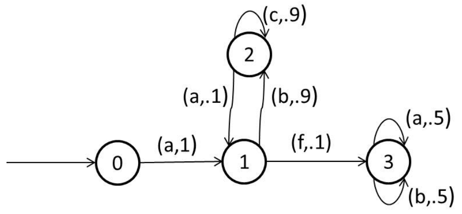

Fig. 1. Stochastic automaton G for Example 1.

$\alpha(x_i, \sigma, x_j) > 0$ and define the language generated by G as $L(G) := \{s \in \Sigma^* \mid \exists x \in X, \alpha(x_0, s, x) > 0\}$.

The observations of events are filtered through an observation mask, $M : \overline{\Sigma} \to \overline{\Delta}$, satisfying $M(\epsilon) = \epsilon$, where Δ is the set of observed symbols. An event σ is said to be unobservable if $M(\sigma) = \epsilon$, and the set of unobservable events is denoted as Σ_{uo} and the set of observable events is then denoted by $\Sigma - \Sigma_{uo}$. The observation mask can be extended from domain Σ to Σ^* inductively as following: $M(\epsilon) = \epsilon$ and $\forall s \in \Sigma^*, \sigma \in \overline{\Sigma}, M(s\sigma) = M(s)M(\sigma)$.

Example 1: Fig. 1 is an example of a stochastic automaton G. The set of states is $X = \{0, 1, 2, 3\}$ with initial state $x_0 = 0$, event set $\Sigma = \{a, b, c, f\}$. A state is depicted as a node, whereas a transition is depicted as an edge between its origin and termination states, with its event name and probability value labeled on the edge. The observation mask M is such that $M(f) = \epsilon$ and otherwise $M(\sigma) = \sigma$.

B. Faulty/nonfaulty Behaviors and Refined Plant

For a stochastic automaton $G = (X, \Sigma, \alpha, x_0)$, its faulty/nonfaulty behaviors can be modeled by partitioning the events set Σ into faulty events $\Sigma_f \subseteq \Sigma$ versus nonfaulty events $\Sigma - \Sigma_f$ where the set of faulty events Σ_f are assumed to be unobservable. Then the overall behaviors of G is given by its generated language $L(G)$, whereas the set of nonfaulty behaviors of G is given by $K = L(G) \cap (\Sigma - \Sigma_f)^*$. The remaining behaviors $L - K$ are called the faulty behaviors. Another approach to describing the faulty/nonfaulty behaviors of a given stochastic automaton G is to specify the nonfaulty behaviors K in form of a *deterministic* automaton $R = (Q, \Sigma, \beta, q)$ such that $L(R) = K$, [11]. Then the refinement of G with respect to R, denoted as *refined plant* G^R, can be used to capture the traces violating the given specification in form of the reachability of a faulty state and is given by $G^R := (Y, \Sigma, \gamma, (x_0, q_0))$, where $Y = X \times \overline{Q}$ and $\overline{Q} = Q \cup \{F\}$, and $\forall (x, \overline{q}), (x', \overline{q}') \in X \times \overline{Q}, \sigma \in \Sigma, \gamma((x, \overline{q}), \sigma, (x', \overline{q}')) = \alpha(x, \sigma, x')$ if the following holds:

$$(\overline{q}, \overline{q}' \in Q \wedge \beta(\overline{q}, \sigma, \overline{q}') > 0)$$
$$\vee (\overline{q} = \overline{q}' = F) \vee \left(\overline{q}' = F \wedge \sum_{q \in Q} \beta(\overline{q}, \sigma, q) = 0 \right),$$

and otherwise $\gamma((x, \overline{q}), \sigma, (x', \overline{q}')) = 0$.

Then it can be seen that the refined plant G^R has the following properties: (1) the generated language of the refined plant G^R is the same as the one generated by G, i.e. $L(G^R) = L(G)$; (2) any trace (system behavior) in $L(G)$ but not in $L(R)$ transitions the refined plant G^R to a faulty state (a state containing F as its second coordinate); (3) the probability of occurrence of each trace in G^R is the same as that in G, i.e., $\sum_{x \in X} \alpha(x_0, s, x) = \sum_{y \in Y} \gamma((x_0, q_0), s, y)$.

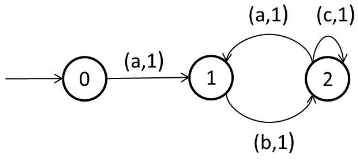

Fig. 2. Deterministic nonfault specification of system G in Fig. 1.

For $y_i, y_j \in Y$ and $\delta \in \Delta$, define the set of traces originating at y_i, terminating at y_j and executing a sequence of unobservable events followed by a single observable event with observation δ as $L_{G^R}(y_i, \delta, y_j) := \{s \in \Sigma^* \mid s = u\sigma, M(u) = \epsilon, M(\sigma) = \delta, \gamma(y_i, s, y_j) > 0\}$. Define $\alpha(L_{G^R}(y_i, \delta, y_j)) := \sum_{s \in L_{G^R}(y_i, \delta, y_j)} \gamma(y_i, s, y_j)$ and denote it as $\mu_{i,\delta,j}$ for short, i.e., it is the probability of all traces originating at y_i, terminating at y_j and executing a sequence of unobservable events followed by a single observable event with observation δ. Also define $\lambda_{ij} = \sum_{\sigma \in \Sigma_{uo}} \gamma(y_i, \sigma, y_j)$ as the probability of transitioning from y_i to y_j while executing a single unobservable event. Then it can be seen that $\mu_{i,\delta,j} = \sum_k \lambda_{ik} \mu_{k,\delta,j} + \sum_{\sigma \in \Sigma : M(\sigma) = \delta} \gamma(y_i, \sigma, y_j)$, where the first term on RHS corresponds to transitioning in at least two steps whereas the second term on RHS corresponds to transitioning in exactly one step. Thus for each $\delta \in \Delta$, given the values $\{\lambda_{ij} | i, j \in Y\}$ and $\{\sum_{\sigma \in \Sigma : M(\sigma) = \delta} \gamma(y_i, \sigma, y_j) | i, j \in Y\}$, all the probabilities $\{\mu_{i,\delta,j} | i, j \in Y\}$ can be found by solving the following matrix equation (see for example [12] for a similar matrix equation):

$$\boldsymbol{\mu}(\delta) \quad = \quad \boldsymbol{\lambda}\boldsymbol{\mu}(\delta) + \boldsymbol{\gamma}(\delta), \tag{1}$$

where $\boldsymbol{\mu}(\delta)$, $\boldsymbol{\lambda}$ and $\boldsymbol{\gamma}(\delta)$ are all $|Y| \times |Y|$ square matrices whose ijth elements are given by $\mu_{i,\delta,j}$, λ_{ij} and $\sum_{\sigma \in \Sigma : M(\sigma) = \delta} \gamma(y_i, \sigma, y_j)$, respectively.

Example 2: For system presented in Fig. 1, the deterministic nonfault specification R is given in Fig. 2. Then the refined plant G^R is shown in Fig. 3. Let the state space of G^R be $Y = \{y_1 = (0, 0), y_2 = (1, 1), y_3 = (2, 2), y_4 = (3, F)\}$. By solving matrix equations (1), we get

$$\boldsymbol{\mu}(a) \quad = \quad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & .05 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & .5 \end{bmatrix}$$
$$\boldsymbol{\mu}(b) \quad = \quad \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & .9 & .05 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & .5 \end{bmatrix}$$
$$\boldsymbol{\mu}(c) \quad = \quad \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & .9 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

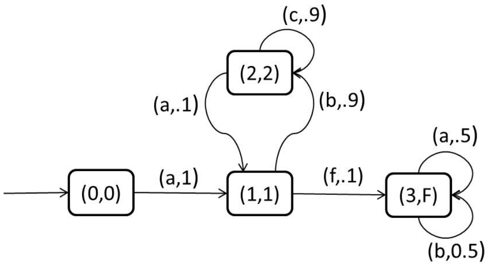

Fig. 3. The refined plant of system G in Fig. 1, when the deterministic nonfault specification R is given in Fig. 2.

III. Stochastic Diagnosability and Online Detection

A. Stochastic Diagnosability of DESs

Let us recall the definition of S-Diagnosability [1] (referred as AA-diagnosability in [5]):

Definition 1: Given a stochastic DES $G = (X, \Sigma, \alpha, x_0)$, deterministic nonfault specification $R = (Q, \Sigma, \beta, q_0)$ with generated languages $L = L(G)$ and $K = L(R)$, (L, K) is said to be Stochastically Diagnosable, or simply S-Diagnosable, if

$$\begin{aligned}&(\forall \tau > 0 \wedge \forall \rho > 0)(\exists n \in \mathbb{N})(\forall s \in L - K)\\ &\qquad Pr(t : t \in L\backslash s, |t| \geq n, P_N(st) > \rho) < \tau,\end{aligned}$$

where $P_N : L - K \to [0, 1]$ is a map that assigns to each faulty trace $s \in L-K$, the probability of s being ambiguous, which is the probability of all nonfaulty traces, conditioned by the fact that ambiguity can only arise from the system traces that produce the same observation as s, and is given by:

$$\begin{aligned}P_N(s) &:= Pr(u \in K | M(u) = M(s))\\ &= \frac{Pr(u \in K : M(u) = M(s))}{Pr(u \in L : M(u) = M(s))}.\end{aligned}$$

Note in the definition of $P_N(s)$, "|" denotes the conditioning operation. Polynomial complexity algorithm for checking S-Diagnosability was also given in [1].

Example 3: By applying algorithm in [1] one can show that system in Fig. 3 is S-Diagnosable. As can be seen, after a fault occurs, and if one continues to observe the system for enough number of transitions, then with high probability two consecutive a or two consecutive b will be observed, resolving the ambiguity that a fault occurred.

Here we present a new characterization of S-Diagnosability which states that the S-Diagnosability is lost if and only if there exists an indistinguishable pair of faulty and nonfaulty traces such that all future observations have identical probability of being faulty versus nonfaulty.

Theorem 1: (L, K) is not S-Diagnosable if and only if:

$$\begin{aligned}&(\exists s \in L - K, s' \in K \text{ s.t. } M(s) = M(s'))(\forall o \in \Delta^*)\\ &\qquad Pr(t : t \in L\backslash s, M(t) = o)\\ &\qquad\qquad = Pr(t : t \in K\backslash s', M(t) = o).\end{aligned}$$

Remark 1: While the definition of S-Diagnosability applies to the set of faulty traces $L - K$, Theorem 1 is symmetric with respect to faulty and nonfaulty traces, and thus suggests that notion of diagnosability can also be defined for nonfaulty traces: $s \in K$ is not diagnosable if and only if there exists $s' \in L - K \cap M^{-1}M(s)$ such that for all future observations $o \in \Delta^*$, $Pr(M^{-1}(o) \cap K\backslash s) = Pr(M^{-1}(o) \cap L\backslash s')$. We denote the set of all non-diagnosable nonfaulty traces as $K^{nd} \subseteq K$. Clearly, for a S-Diagnosable system, $K^{nd} = \emptyset$.

B. Computation of Likelihood of No-fault

When the system executes a trace $s \in L$, an observation $o = M(s)$ is received by a fault detector. In order to issue a fault-decision versus no-decision for the observation $o = M(s)$, we propose the detector compute the likelihood of no-fault, and issue a fault-decision if this likelihood of no-fault is below a suitable threshold, and otherwise issue no-decision. In this subsection we present how this likelihood can be recursively computed. With a slight abuse of notation, we denote the no-fault likelihood function $P_N : M(L) \to [0, 1]$ and define it to be the conditional probability of nonoccurrence of a fault following any observation $o \in M(L)$:

$$P_N(o) \quad := \quad \frac{Pr(u \in K : M(u) = o)}{Pr(u \in L : M(u) = o)}.$$

Note that $P_N(o)$ is the probability of nonfaulty traces conditioned by the fact that ambiguity can only arise from the system traces that produce the observation o. In order to recursively compute P_N we proceed as follows. For a given refined plant G^R whose state space is partitioned into nonfaulty states versus faulty states, we define a nonfault indication binary column vector $I_{nf} \in \{0, 1\}^{|Y| \times 1}$, where an entry of 1 indicates a nonfaulty state. Also define state distribution vector $\boldsymbol{\pi} : M(L) \to [0, 1]^{1 \times |Y|}$, i.e., for each $o \in M(L)$, $\boldsymbol{\pi}(o)$ is the state distribution of G^R following the observation o. Then $\boldsymbol{\pi}(\cdot)$ is recursively computed as follows: $\boldsymbol{\pi}(\epsilon) = [1, 0, \ldots, 0]$, and for any $o \in M(L), \delta \in \Delta$,

$$\boldsymbol{\pi}(o\delta) \quad = \quad \frac{\boldsymbol{\pi}(o)\boldsymbol{\mu}(\delta)}{||\boldsymbol{\pi}(o)\boldsymbol{\mu}(\delta)||},$$

where $\boldsymbol{\mu}(\delta)$ is computed by solving matrix equations (1), and $\| \cdot \|$ is simply the sum of all vector elements. Then for an observation o, $P_N(o)$ is simply given by

$$P_N(o) \quad = \quad \boldsymbol{\pi}(o) I_{nf},$$

where note that $\boldsymbol{\pi}(o)$ and hence also $P_N(o)$ are recursively computed.

Example 4: In the system of Fig. 3, the indication vector is given as $I_{nf} = [1, 1, 1, 0]^T$, and the state distribution vector is initialized as $\boldsymbol{\pi}(\epsilon) = [1, 0, 0, 0]$. If $o = aba$, then $P_N(o) = 0.783$; if $o = ababc$, then $P_N(o) = 1$; if $o = abaa$, then $P_N(o) = 0$.

C. Online Detection Scheme

For issuing online detection decision, we propose a detector, $D : M(L) \to F \cup \{\epsilon\}$ that for each observation in $M(L)$ issues either a "fault (F)" decision or "no-decision (ϵ)" by comparing the likelihood of no-fault to a suitable threshold, as follows:

$$\forall o \in M(L), [D(o) = F] \Leftrightarrow [\exists \overline{o} \leq o : P_N(\overline{o}) \leq \rho_D], \quad (2)$$

where ρ_D is the detection threshold, appropriately chosen to meet the desired FA rate requirement. Note by definition, if a detection decision is F, then it remains F for all future observations, i.e., the detector "does not change its mind", which is expected for the case of permanent faults.

Note a *false alarm* occurs if the detector D issues F while the refined plant is in a nonfaulty state; and dually a *missed detection* occurs if the detector D fails to issue a F decision within an appropriate delay bound n_D after the occurrence of a fault. In other words, letting P_D^{md} and P_D^{fa} denote the MD and FA rates respectively of a detector D, then

$$\begin{aligned} P_D^{md} &:= Pr(st \in L - K : s \in L - K, \\ &\qquad |t| \geq n_D, P_N(M(st)) > \rho_D), \quad (3) \\ P_D^{fa} &:= Pr(s \in K : P_N(M(s)) \leq \rho_D). \quad (4) \end{aligned}$$

Example 5: For the refined plant of Fig. 3 which is S-Diagnosable, suppose we set the threshold $\rho_D = 0.8$. Then any nonfaulty trace in $a(bc^+a)^*ba \subset K$ will be false-alarmed ($P_N(ababa) = 0.783 < \rho_D$), and thus, $P_D^{fa}|_{\rho_D=0.8} = Pr(u \in a(bc^+a)^*ba) = 47.37\%$. On the other hand if we set $\rho_D = 0.5$, then any nonfaulty trace in $a(bc^+a)^*baba \subset K$ will be false-alarmed ($P_N(ababa) = 0.488 < \rho_D$), and thus, $P_D^{fa}|_{\rho_D=0.5} = Pr(u \in a(bc^+a)^*baba) = 4.26\%$. Now supposing that 4.26% FA rate is acceptable, we fix the detection threshold ρ_D to 0.5. If the detection delay bound is set to be $n_D = 3$, then any faulty trace $s \in a(bc^+a)^*fbab \in L - K$ will be miss-detected and thus the MD rate is given by $P_D^{md}|_{\rho_D=0.5, n_D=3} = 6.58\%$. On the other hand if the detection delay bound is set to be $n_D = 4$, then any faulty trace $s \in L - K$ can be detected, i.e., $P_D^{md}|_{\rho_D=0.5, n_D=4} = 0$.

The following theorem provides insight into the significance of the S-Diagnosability property for the purpose of online fault detection, by showing its necessity and sufficiency for the existence of an online detector that can achieve any desired levels of MD and FA rates.

Theorem 2: (L, K) is S-Diagnosable if and only if for any FA rate requirement $\phi > 0$ and MD rate requirement $\tau > 0$, there exist a detection threshold $\rho_D > 0$ and a delay bound n_D such that $P_D^{fa} \leq \phi$ and $P_D^{md} \leq \tau$.

D. Algorithms for ρ_D and n_D

In this subsection we provide algorithms for computing the parameters ρ_D and n_D so as to achieve the desired level of MD and FA rates. In order to compute detection threshold ρ_D for a given FA rate requirement ϕ, Algorithm 1 constructs an extended observer tree that for each observation sequence estimates the states, with the estimate labeled by the observation, and each state in the estimate labeled by the probability of reaching it. These probability labels are then used to compute the probability P_N for each observation, or equivalently, of each node of the extended observer tree. The tree extends to a depth so that if no detection decision are made for any of the nodes (equivalently, corresponding unique observations) in the tree, then the FA rate caused by the detection decisions at the future successors is upper bounded by the desired rate ϕ. The existence of such a depth is guaranteed by Theorem 3, and to ensure no detection decision for any of the nodes in T, we simply choose the detection threshold to be smaller than the minimum P_N value among all nodes of T (recall by (2) that a detection decision is only issued when the P_N value falls below the threshold).

Algorithm 1: For a given refined plant G^R and a FA rate requirement ϕ, do the following:

1) Identify all the states in $X \times Q$ from which a faulty state in G^R is reachable, and denote this set of states as Y_1 (these are nonfaulty states from where faulty states are reachable). Identify all the states in $X \times Q - Y_1$ that appear as the second coordinates of states in bi-closed SCCs that violate the condition (III) in [1, Theorem 4] and denote this set of states as Y_2 (these are nonfaulty nondiagnosable states), and also identify $Y_3 = X \times Q - Y_1 - Y_2$ (there are nonfaulty diagnosable states).
2) Iteratively construct an extended observer tree T with set of nodes, $\overline{Z} = Z \times M(L)$, where $Z = 2^{((X \times \overline{Q}) \times (0,1])}$, and the depth of tree grows by 1 in each iteration until the stopping criterion is satisfied—see below. Then each node of T is of the form $\overline{z} = (z, o(\overline{z}))$, where $z = \{((x_i, \overline{q}_i), p_i)\} \subseteq (X \times \overline{Q}) \times (0, 1]$ and $o(\overline{z}) \in M(L)$, and each node $\overline{z}$ corresponds to a unique observation $o(\overline{z})$. The tree T is rooted at $\overline{z}_0 = \{((0, 0), 1), \epsilon\}$. $\overline{z}_2 \in \overline{Z}$ is a δ-child ($\delta \in \Delta = M(\Sigma) - \{\epsilon\}$) of $\overline{z}_1 \in \overline{Z}$ if and only if $o(\overline{z}_2) = o(\overline{z}_1)\delta$ and for every $((x_2, \overline{q}_2), p_2) \in z_2$, it holds that $p_2 = \sum_{((x_1,\overline{q}_1),p_1) \in z_1} \sum_{s \in \Sigma^* : M(s) = \delta} p_1 \times \gamma((x_1, \overline{q}_1), s, (x_2, \overline{q}_2))$. *It can be seen that $((x_2, \overline{q}_2), p_2)$ is included in z_2 if and only if p_2 is the probability of reaching $(x_2, \overline{q}_2)$ following the observation $o(\overline{z}_2)$.*

For each node $\overline{z} = (z, o(\overline{z}))$, define:

$$P_N(\overline{z}) \quad := \quad \frac{\sum_{((x,\overline{q}),p) \in \overline{z}, \overline{q} \neq F} p}{\sum_{((x,\overline{q}),p) \in \overline{z}} p}.$$

(Note here P_N is defined over the states of the extended observer T, while earlier it was defined over the observed traces.) Then $P_N(\overline{z}) = P_N(o(\overline{z}))$ is the conditional probability of no-fault given the observation $o(\overline{z})$. The tree is terminated at a uniform depth so the set of leaf nodes $\overline{Z}_m \subseteq \overline{Z}$ satisfy:

- $(\overline{z}, \overline{z}' \in \overline{Z}_m) \Rightarrow (|o(\overline{z})| = |o(\overline{z}')| =: d_1)$ (each terminal node is reached after the same number of observations, which guarantees the uniformity of the depth of T, which we denote as d_1), and
- $(((x, \overline{q}), p) \in \overline{z} \cap Y_2 \times (0, 1], \overline{z} \in Z_m) \Rightarrow (\exists \overline{z}' \in \overline{Z})(o(\overline{z}') \leq o(\overline{z}), |o(\overline{z})| - |o(\overline{z}')| > |X \times \overline{Q}|, P_N(\overline{z}') = P_N(\overline{z}))$ (if a terminal node contains

an element in Y_2, then it must be part of a cycle in which probability of no-fault has stopped decreasing, and so there is no gain of further extending the tree since by choosing the threshold to be less than the converged value, we can ensure that no decisions are made and so no false alarm would occur for the observation leading to this terminal node), and

- $\sum_{\overline{z}\in\overline{Z}_m}\sum_{((x,\overline{q}),p)\in\overline{z}:(x,\overline{q})\in Y_1} p + \sum_{\overline{z}\in\overline{Z}_m:P_N(\overline{z})\leq\rho_{\min}}\sum_{((x,\overline{q}),p)\in\overline{z}:(x,\overline{q})\in Y_3} p < \phi$, where $\rho_{\min} := \min_{\overline{z}\in\overline{Z}:P_N(\overline{z})\neq 0} P_N(\overline{z})$ (for states in Y_1 contained in terminal nodes, their added probabilities (i.e., the first term on the LHS) equals $Pr(K_1 \cap \Sigma^{>d_1})$, which upper bounds the FA rate of their successors (see proof of Theorem 2); for the states in Y_3 contained in the terminal nodes having $P_N \leq \rho_{\min}$, their added probabilities (i.e., the second term on the LHS) equals $Pr(s \in K_3 \cap \Sigma^{>d_1} : P_N(s) \leq \rho_{\min})$, which upper bounds the FA rate of their successors (see proof of Theorem 2); by our selection of threshold ρ_D—see step 3 below, none of the nodes in T has decision and hence no false alarms, so the overall FA rate is given by the rate of false alarms of the future successors of the terminal nodes, which is required to be less than ϕ).

3) Return any $\rho_D < \rho_{\min}$. (Note that with this choice of ρ_D, all observations included in T will have no detection decisions (and so no false alarms either), and only their extensions can have detection decisions (some of which may be false alarms). But by construction, the probability of those extensions is upper bounded by ϕ, as desired.)

The following theorem guarantees the correctness of Algorithm 1.

Theorem 3: There exists $d_1 \in \mathbb{N}$ such that Algorithm 1 terminates with tree depth d_1 and returns a threshold ρ_D under which the overall FA rate is upper bounded by ϕ.

Now that we have provided an algorithm to compute the detection threshold ρ_D that meets the FA rate ϕ, we next present an algorithm to compute the delay bound n_D to satisfy the given MD rate τ. In order to compute delay bound n_D, Algorithm 2 constructs a refined version of the extended observer tree that for each observation sequence estimates the states and their probabilities, with the refinement that keeps track of the number of post-fault transitions executed for each state in the estimated state-set. The tree extends to a depth so that if no missed detections occur for any of the nodes in the tree, then the MD rate caused by the future successors is upper bounded by the desired rate τ. For S-Diagnosable systems, the existence of such a depth is guaranteed by Theorem 4, and to ensure no missed detection for any of the nodes in T, we simply choose n_D to be greater than the maximum number of post fault transitions among all nodes of T (recall from (3) that a missed detection occurs only if a fault remains undetected beyond n_D number of transitions).

Algorithm 2: For a given refined plant G^R, a detection threshold ρ_D and a MD rate requirement τ, do the following:

1) Iteratively construct a refined extended observer tree T with set of nodes, $\overline{Z} = Z \times M(L)$, where $Z = 2^{((X\times\overline{Q})\times(0,1]\times\mathbb{N})}$ ($\mathbb{N} = \{0, 1, 2, \dots\}$), and the depth of T grows by 1 in each iteration until the stopping criterion is satisfied—see below. As in Algorithm 1, each node of T is of the form $\overline{z} = (z, o(\overline{z}))$, where $z = \{((x_i, q_i), p_i, n_i)\} \subseteq (X \times \overline{Q}) \times (0, 1] \times \mathbb{N}$ and $o(\overline{z}) \in M(L)$. The tree T is rooted at $\overline{z}_0 = \{(((0, 0), 1, 0), \epsilon\}$. $\overline{z}_2 \in \overline{Z}$ is a δ-child ($\delta \in \Delta = M(\Sigma) - \{\epsilon\}$) of $\overline{z}_1 \in \overline{Z}$ if and only if $o(\overline{z}_2) = o(\overline{z}_1)\delta$, and for every $((x_2, \overline{q}_2), p_2, n_2) \in z_2$, it holds that $p_2 = \sum_{((x_1,\overline{q}_1),p_1,n_1)\in z_1} \sum_{s\in\Sigma^*:M(s)=\delta,\#\text{post-fault}(s,(x_1,\overline{q}_1))+n_1=n_2} p_1 \times \gamma((x_1, \overline{q}_1), s, (x_2, \overline{q}_2))$. Here "#post-fault" counts the number of events in s beyond a fault as follows: if $\overline{q}_1 = F$, it returns the value $|s|$, and otherwise it returns the number of transitions executed in s after a faulty state is reached. *It can be seen that $((x_2, \overline{q}_2), p_2, n_2)$ is included in z_2 if and only if p_2 is the probability of reaching x_2 following the observation $o(\overline{z}_2)$ and n_2 is the number the post-fault transitions executed.*

   For each node $\overline{z} = (z, o(\overline{z}))$, define:

$$P_N(\overline{z}) \quad := \quad \frac{\sum_{((x,\overline{q}),p,n)\in z,\overline{q}\neq F} p}{\sum_{((x,\overline{q}),p,n)\in z} p}.$$

   A branch of the tree is terminated if a detection decision has been made (P_N value smaller than ρ_D), and the tree itself is terminated at a uniform depth so the set of leaf nodes $\overline{Z}_m \subseteq \overline{Z}$ satisfy:

   - $P_N(\overline{z}) \leq \rho_D$ (for these nodes detection decision can be issued, implying these nodes will have no missed detections), or
   - $\sum_{\overline{z}\in\overline{Z}_m:P_N(\overline{z})>\rho_D}\sum_{((x,\overline{q}),p,n)\in\overline{z}:(x,\overline{q})\in Y_1\vee\overline{q}=F} p < \tau$ (for these nodes, no detection decision will be issued, and by the choice of n_D in step 2 below there is no missed detection yet; so their added probabilities upper bounds the MD rate of their future successors, and the stopping criterion requires that this to be below the desired value τ).

2) Return any $n_D > \max_{((x,\overline{q}),p,n)\in z,\overline{z}\in\overline{Z}} n$, and let d_2 denote the depth of tree T. Note that with this choice of n_D all faulty traces, whose observations are included in T, are not miss-detected. So clearly that the MD rate P_D^{md} is upper bounded by $\overline{P_D^{md}}$ given by:

$$\overline{P_D^{md}} \quad := \sum_{\overline{z}\in\overline{Z}_m:P_N(\overline{z})>\rho_D} \quad \sum_{((x,\overline{q}),p,n)\in\overline{z}:(x,\overline{q})\in Y_1\vee\overline{q}=F} p \tag{5}$$

The following theorem guarantees the correctness of Algorithm 2.

Theorem 4: For S-Diagnosable systems, there exists $d_2 \in \mathbb{N}$ such that Algorithm 2 terminates with tree depth d_2 and returns a delay bound n_D under which the overall MD rate is upper bounded by τ.

Note that if the system is not S-Diagnosable, the termination of Algorithm 2 is not guaranteed. A modified version of the algorithm guaranteeing termination is presented below in Algorithm 3 that finds an upper bound for the minimum achievable MD rate for a given detection threshold.

E. Non-S-Diagnosable Systems

Theorem 2 guarantees arbitrary performance level could be achieved by detector D if the system is S-Diagnosable; this may not be true when the system is not S-Diagnosable. For given ϕ and τ, let ρ_D be chosen so that $P_D^{fa} \leq \phi$, and let $S_D^{nd} \subseteq L - K$ be the set of non-diagnosable faulty traces for which there exists a MD rate $\tau' > 0$ such that the condition $Pr_D^{md}(S_D^{nd}) = Pr(st : s \in S_D^{nd}, t \in L \backslash s, |t| \geq n_D, P_N(st) > \rho_D) < \tau'$ is not satisfied by any $n_D \in \mathbb{N}$. Then for the traces in $(L-K) - S_D^{nd}$ there exists a detection delay bound n_D so that $\forall s \in (L-K) - S_D^{nd}$, $Pr(t : t \in L \backslash s, |t| \geq n_D, P_N(st) > \rho_D) < \tau'$, and so the overall MD rate is upper bounded by:

$$\begin{aligned} P_D^{md} &= \sum_{s \in L-K} Pr_D^{md}(s) Pr(s) \\ &< \tau' Pr(L - K - S_D^{nd}) + Pr_D^{md}(S_D^{nd}) \\ &\leq \tau' + Pr_D^{md}(S_D^{nd}). \end{aligned}$$

Thus for non-S-Diagnosable systems, while any desired FA rate $\phi > 0$ can be always achieved by an appropriate choice of $\rho_D > 0$, a MD rate $\tau > 0$ can only be achieved if $\tau' + Pr_D^{md}(S_D^{nd}) \leq \tau$. Since n_D can be chosen to make τ' arbitrarily small, a MD rate $\tau > 0$ can be achieved if and only if $Pr_D^{md}(S_D^{nd}) < \tau$. This is captured in the following theorem, which generalizes Theorem 2 to the case of non-S-Diagnosable systems.

Theorem 5: Given a stochastic, nonfault specification-refined plant G^R with generated language L and nonfault behavior K, FA rate requirement $\phi > 0$ and MD rate requirement $\tau > 0$, there exists a detection threshold $\rho_D > 0$ such that $P_D^{fa} \leq \phi$, and for this detection threshold there exists a detection delay bound n_D such that $P_D^{md} \leq \tau$ if and only if $Pr_D^{md}(S_D^{nd}) \leq \tau$, where $S_D^{nd} \subseteq L - K$ is the set of faulty traces for which there exists $\tau' > 0$ such that the condition $Pr(st : s \in S_D^{nd}, t \in L \backslash s, |t| \geq n_D, P_N(st) > \rho_D) < \tau'$ is not satisfied by any $n_D \in \mathbb{N}$.

Note that for fixed ρ_D, $Pr_D^{md}(S_D^{nd})$ is also fixed and serves as a lower bound for MD rate that the detection scheme can achieve. Next we present a variant of Algorithm 2 that for a fixed threshold ρ_D computes an upper bound for $Pr_D^{md}(S_D^{nd})$.

Algorithm 3: For a given refined plant G^R and a threshold ρ_D, do the following:

1) Iteratively construct a refined extended observer tree T as in the step 1 of Algorithm 2 by adding an extra level of depth in each iteration;
2) For each depth of the tree T, set $n_D = 1 + \max_{((x,\overline{q}),p,n) \in z, \overline{z} \in \overline{Z}} n$ and compute an upper bound $\overline{P_D^{md}}$ for MD rate P_D^{md} according to (5);
3) If the upper bound $\overline{P_D^{md}}$ doesn't decrease while n_D computed in step 2 gets doubled over any two iteration steps (not necessarily consecutive), stop and return this upper bound.

IV. Conclusion

In this paper, the problem of online fault diagnosis for stochastic DESs was studied. An online detector based on recursive likelihood computation was proposed, whose existence for achieving any arbitrary performance requirement was shown to be equivalent to the S-Diagnosability property. Algorithms for computing the detector parameters of detection threshold and delay bound so as to achieve a given performance requirement of false alarm and missed detection rates were presented, using a proposed procedure for constructing an extended observer. It was also shown that our detection strategy works for S-Diagnosable as well as non-S-Diagnosable systems in the same manner. For S-Diagnosable systems it is possible to achieve arbitrary performance for FA and MD rates, while for a non-S-Diagnosable system an arbitrary performance is achievable only for the FA rate, whereas a lower bound exists for the achievable MD rate that is a function of the FA rate, and increases as FA rate is decreased. A variant of the algorithm for the S-Diagnosable case was used to compute an upper bound for the minimum achievable missed detection rate for a non-S-Diagnosable system.

References

[1] J. Chen and R. Kumar, "Polynomial test for stochastic diagnosability of discrete event systems," *IEEE Trans. Auto. Sci. Eng.*, Apr. 2013, DOI=10.1109/TASE.2013.2251334.

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

[3] S. Jiang, Z. Huang, V. Chandra, and R. Kumar, "A polynomial algorithm for testing diagnosability of discrete-event systems," *IEEE Trans. Autom. Control*, vol. 46, no. 8, pp. 1318–1321, Aug. 2001.

[4] T.-S. Yoo and S. Lafortune, "Polynomial-time verification of diagnosability of partially observed discrete-event systems," *IEEE Trans. Autom. Control*, vol. 47, no. 9, pp. 1491–1495, Sep. 2002.

[5] D. Thorsley and D. Teneketzis, "Diagnosability of stochastic discrete-event systems," *IEEE Trans. Autom. Control*, vol. 50, no. 4, pp. 476–492, Apr. 2005.

[6] W.-C. Lin, H. E. Garcia, and T.-S. Yoo, "A diagnoser algorithm for anomaly detection in DEDS under partial and unreliable observations: characterization and inclusion in sensor configuration optimation," *Discrete Event Dyn. Syst.*, vol. 23, no. 1, pp. 61–91, Mar. 2013.

[7] J. Lunze, "Fault diagnosis of discretely controlled continuous systems by means of discrete-event models," *Discrete Event Dyn. Syst.*, vol. 18, no. 2, pp. 181–210, 2008.

[8] S. Jiang and R. Kumar, "Failure diagnosis of discrete-event systems with linear-time temploral logic specifications," *IEEE Trans. Autom. Control*, vol. 49, no. 6, pp. 934–945, Jun. 2004.

[9] J. Chen and R. Kumar, "Decentralized failure diagnosis of stochastic discrete event systems," in *Proc. 9th IEEE Int. Conf. Autom. Sci. Eng.*, Madison, WI, Aug. 2013.

[10] V. K. Garg, R. Kumar, and S. I. Marcus, "A probabilistic language formalism for stochastic discrete-event systems," *IEEE Trans. Autom. Control*, vol. 44, no. 2, pp. 280–293, Feb. 1999.

[11] W. Qiu and R. Kumar, "Decentralized failure diagnosis of discrete event systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Human*, vol. 36, no. 2, pp. 384–395, Mar. 2006.

[12] X. Wang and A. Ray, "A language measure for performance evaluation of discrete-event supervisory control systems," *Applied Math. Modelling*, vol. 28, no. 9, pp. 817–833, Sep. 2004.