A Probabilistic Test for A-Diagnosability of Stochastic Discrete-Event Systems With Guaranteed Error Bound

Jun Chen, *Senior Member, IEEE*

***Abstract*—This letter investigates the failure diagnosability of stochastic discrete-event systems (DES). Specifically, the A-Diagnosability (proposed by Thorsley et al., 2005) is studied, which requires every failure to be stochastically diagnosable with arbitrary probability and within a certain delay bound. The verification of A-Diagnosability was later shown to be PSPACE-Complete, and a polynomial testing algorithm likely does not exist. This letter fills this gap by providing a new necessary and sufficient condition for checking A-Diagnosability of stochastic DES, based on which a probabilistic test is also proposed. The complexity of the proposed algorithm is polynomial in the number of system states and events, with a sacrifice that the proposed test will also incur certain test errors. Furthermore, the balance between computing complexity and probability of test error is calibratable through a hyper-parameter. Several working examples are provided to illustrate the proposed verification condition and probabilistic test.**

***Index Terms*—Discrete event systems, fault diagnosis, stochastic systems, randomized algorithms.**

I. Introduction

DETECTING system failures is critical in many applications including automotive systems [1] and power systems [2]. To ensure system operation, system failures need to be detected with acceptable accuracy and within a tolerable delay bound. The problem of fault diagnosis has been widely researched for discrete-event systems (DES) [3], [4], [5], [6], [7], [8], [9], [10], [11]. The notion of *Diagnosability* of DES was introduced in [5], which requires any faulty trace to be detectable within a finite delay. The verification of Diagnosability was addressed in [3], [4], together with other extensions in distributed setting [9] and decentralized setting [12]. See [13] for a recent survey for diagnosis in DES.

Fault diagnosis has later been studied for stochastic DES [14], [15], [16], [17], [18], [19], [20], [21]. In particular, the notion of *A-Diagnosability* for stochastic DES was introduced in [18], which requires that given any error bound τ, there must exist a delay bound n such that the set of undetectable faulty traces that are longer than n occurs with a probability smaller than τ. In other words, A-Diagnosability does not require every fault to be detectable within a bounded delay (which is required by Diagnosability) but guarantees that the probability of detecting the fault for sure converges to 1 as one keeps observing the system for a sufficiently long time. Since the seminal work of [18], several variations of A-Diagnosability, such as safe diagnosability [21], robus diagnosability [22], and co-diagnosability [23], were also investigated. The A-Diagnosability problem was cast into a probabilistic logic problem in [24]. Furthermore, the prognosis of stochastic DES has been reported in [25], [26], while the control of stochastic DESs was examined in [27], [28].

Verification for A-Diagnosability was also studied in [18], which are based on certain structural properties of a diagnoser and therefore require *exponential* complexity in the number of system states. In fact, it was later shown in [19] that verifying A-Diagnosability is PSPACE-Complete, and therefore a polynomial test algorithm likely does not exist, making the verification of A-Diagnosability for large systems practically impossible. To fill this gap, this letter proposes a probabilistic test for verifying A-Diagnosability. First, a new necessary and sufficient condition for checking A-Diagnosability is developed, which requires that every recurrent faulty state is either reachable unambiguously or the generated masked language from it is not a subset of the generated masked language of its ambiguous nonfaulty state. Verifying such conditions is, unfortunately, still exponential in the number of system states due to nondeterminism, but it provides a direction to conduct a probabilistic test with polynomial complexity and guarantee error bound.

Second, a probabilistic and polynomial test algorithm is developed, which randomly selects N extensions from a recurrent ambiguous faulty state and checks if the fault can be detected by the sampled extensions. If all of the N randomly selected extensions are not detectable, then the system is determined to be not A-Diagnosable. This is repeated for all recurrent ambiguous faulty states. The proposed test algorithm can return correct test results when the system is not A-Diagnosable, while for an A-Diagnosable system, there is a possibility of returning an incorrect test result since only

Manuscript received 13 April 2023; revised 5 June 2023; accepted 26 June 2023. Date of publication 28 June 2023; date of current version 14 July 2023. Recommended by Senior Editor A. P. Aguiar.

The author is with the Department of Electrical and Computer Engineering, Oakland University, Rochester, MI 48374 USA (e-mail: junchen@oakland.edu).

Digital Object Identifier 10.1109/LCSYS.2023.3290476

a subset of extensions are checked. Furthermore, the number of sampled extensions, N, influences both computing complexity and probability of test error and therefore can be treated as a hyperparameter to balance computation and test accuracy.

The contribution of this letter is summarized as follows.

1) A new necessary and sufficient condition for verifying A-Diagnosability for stochastic DES is developed.
2) A probabilistic test for verifying A-Diagnosability is proposed, which has a polynomial complexity. The proposed probabilistic test can correctly verify a non-A-Diagnosable system and for A-Diagnosable system the error bound is guaranteed.

II. Notation and Preliminary

A. Stochastic Discrete-Event Systems

For an event set Σ, let $\overline{\Sigma} := \Sigma \cup \{\epsilon\}$ denote the set of events plus ϵ, the "no-event". Let Σ^* and Σ^+ denote the set of all finite length event sequences over Σ, including and excluding ϵ respectively, i.e., $\Sigma^+ = \Sigma^* - \{\epsilon\}$. A member of Σ^* is called a *trace*. Denote as $s \in pr(t)$ if $s \in \Sigma^*$ is a prefix of $t \in \Sigma^*$, and use $|s|$ to denote the number of events in s (the length of s). A subset of Σ^* is called *language*. 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* $G = (X, \Sigma, \alpha, x_0)$, where X is the set of states, Σ is the finite set of events, $x_0 \in X$ is the initial state, and $\alpha : X \times \Sigma \times X \to [0, 1]$ is the transition probability function [29]. G is said to be non-stochastic if $\alpha : X \times \Sigma \times X \to \{0, 1\}$, and a non-stochastic DES is said to be deterministic if $\forall x \in X, \sigma \in \Sigma, \sum_{x' \in X} \alpha(x, \sigma, x') \in \{0, 1\}$. The transition probability function α can be extended from domain $X \times \Sigma \times X$ to $X \times \Sigma^* \times X$ in a natural way. Define the language generated by G as $L(G) := \{s \in \Sigma^* : \exists x \in X, \alpha(x_0, s, x) > 0\}$. The events are observed through an observation mask, $M : \overline{\Sigma} \to \overline{\Delta}$, satisfying $M(\epsilon) = \epsilon$, where Δ is the set of observable symbols and $\overline{\Delta} := \Delta \cup \{\epsilon\}$. An event σ is said to be unobservable if $M(\sigma) = \epsilon$; the set of unobservable events is denoted by Σ_{uo} and the set of observable events is given by $\Sigma_o = \Sigma - \Sigma_{uo}$. The observation mask can be extended from domain Σ to Σ^* in a natural way.

Example 1: Fig. 1(a) is an example of a stochastic automaton G, which was also studied in [16]. 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 $M(\sigma) = \sigma$ for $\sigma \in \{a, b, c\}$.

A *component* $C = (X_C, \alpha_C)$ of G is a "subgraph" of G, i.e., $X_C \subseteq X$ and $\forall x, x' \in X_C$ and $\sigma \in \Sigma$, $\alpha_C(x, \sigma, x') = \alpha(x, \sigma, x')$, whenever the latter is defined. C is said to be a *strongly connected component* (SCC) or *irreducible* if $\forall x, x' \in X_C$, $\exists s \in \Sigma^*$ such that $\alpha_C(x, s, x') > 0$. An SCC C is said to be *closed* if for each $x \in X_C$, $\sum_{\sigma \in \Sigma} \sum_{x' \in X_C} \alpha_C(x, \sigma, x') = 1$. In other words, if an SCC C is closed, then the probability of staying in C is 1 once entering it. The states which belong to a closed SCC are *recurrent states*.

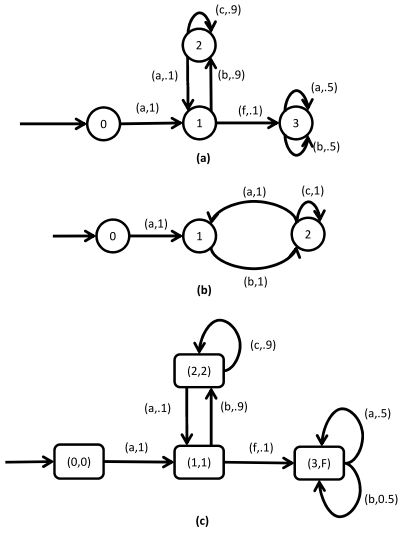

Fig. 1. (a) Stochastic automaton G; (b) Deterministic nonfault specification R; (c) Refined plant G^R.

For a stochastic DES $G = (X, \Sigma, \alpha, x_0)$, its nonfaulty behaviors are specified in the form of a *deterministic* automaton $R = (Q, \Sigma, \beta, q_0)$ such that $L(R) = K$ is the set of nonfaulty traces. Then the remaining traces $L - K$ are called faulty behaviors. The refinement of G with respect to R, denoted as G^R, can be used to capture the faulty traces in the form of the reachability of a faulty state (with the second coordinate labeled with F) and is defined by $G^R := (X \times \overline{Q}, \Sigma, \gamma, (x_0, q_0))$, where $\overline{Q} = Q \cup \{F\}$, 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 $(\overline{q}, \overline{q}' \in Q \wedge \beta(\overline{q}, \sigma, \overline{q}') > 0) \vee (\overline{q} = \overline{q}' = F) \vee (\overline{q}' = F \wedge \sum_{q \in Q} \beta(\overline{q}, \sigma, q) = 0)$ holds, and otherwise $\gamma((x, \overline{q}), \sigma, (x', \overline{q}')) = 0$. In other words, G^R is a composition of G and R such that the generated language of G^R is the same as G with the probability of each trace being the same in G and in G^R. Furthermore, any faulty trace will transition G^R to a faulty state with the second coordinate labeled with F.

Example 2: For the system presented in Fig. 1(a) and discussed in Example 1, suppose the deterministic nonfault specification R is given in Fig. 1(b). Then the refined plant G^R is shown in Fig. 1(c). For G^R, it is apparent that G^R has only two SCCs: C_1 consisting of $(3, F)$ and the two self-loop transitions, and C_2 consisting of $(1, 1)$ and $(2, 2)$ and transitions among them. Moreover, C_1 is a closed SCC while C_2 is not closed.

B. A-Diagnosability of Stochastic DES

The objective of the diagnosability problem is to characterize the conditions under which the occurrence of a faulty trace $s \in L - K$ can be detected within an uniformly bounded delay. The definition of *A-Diagnosability* for stochastic DES requires that given any error bound τ, there must exist a delay bound n such that the set of undetectable faulty traces that are longer than n occurs with a probability smaller than τ. Definition 1 below formally defines A-Diagnosability and is rephrased from [18].

Definition 1 [18]: Given a stochastic DES G, deterministic nonfault specification R with generated languages $L = L(G)$

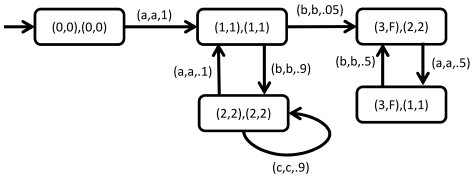

Fig. 2. Testing automaton T for the system in Fig. 1.

and $K = L(R)$, (G, R) is said to be A-Diagnosable, if

$$(\forall \tau > 0)(\exists n \in \mathbb{N})(\forall s \in L - K) \\ Pr(t : t \in L\backslash s, |t| \geq n, Pr_{\text{amb}}(st) > 0) < \tau,$$

where $Pr_{\text{amb}} : L - K \to [0, 1]$ is given by $Pr_{\text{amb}}(s) = Pr(u \in K : M(u) = M(s))/Pr(u \in L : M(u) = M(s))$ and $Pr(\cdot)$ is the probability notation denoting the probability of a given set.

Example 3: For the system presented in Fig. 1 and discussed in Examples 1 and 2. After a faulty trace is executed and G^R transitions into $(3, F)$, some of its extensions such as $(ba)^*$ are ambiguous with another nonfaulty trace, while all extensions with two consecutive a or b are unambiguous. Moreover, the probability of ambiguous extensions for sure converges to 0 when their lengths increase. According to Definition 1, the system in Fig. 1 is A-Diagnosable.

Definition 2 [30]: For a given stochastic automaton $G = (X, \Sigma, \alpha, x_0)$ and a deterministic nonfault specification $R = (Q, \Sigma, \beta, q_0)$, a testing automaton $T = G^R \times G^R$ can be constructed such that in each step, the first copy of G^R takes lead by executing a sequence of unobservable events followed by a single observable event, whereas the second copy responds by executing ambiguous nonfaulty traces. This automaton is denoted as $T = (Z, \Sigma \times \Sigma, \delta, z_0)$, where

- $Z = (X \times \overline{Q}) \times (X \times \overline{Q})$;
- $z_0 = ((x_0, q_0), (x_0, q_0))$ is the initial state;
- $\delta : Z \times \Sigma \times \Sigma \times Z \to [0, 1]$ is defined as: $\forall((x_1, \overline{q}_1), (x_2, \overline{q}_2)), ((x_1', \overline{q}_1'), (x_2', \overline{q}_2')) \in Z$ and $(\sigma_1, \sigma_2) \in \Sigma \times \Sigma$, $\delta(((x_1, \overline{q}_1), (x_2, \overline{q}_2)), (\sigma_1, \sigma_2), ((x_1', \overline{q}_1'), (x_2', \overline{q}_2'))) = \alpha(L_{G^R}((x_1, \overline{q}_1), \sigma_1, (x_1', \overline{q}_1'))) \times \frac{\alpha(L_{G^R}((x_2, \overline{q}_2), \sigma_2, (x_2', \overline{q}_2')))}{\sum_{(x_2', \overline{q}_2') \in X \times Q} \alpha(L_{G^R}((x_2, \overline{q}_2), M(\sigma_2), (x_2', \overline{q}_2')))}$ if $(\sigma_1 \in \Sigma - \Sigma_{uo}) \wedge (M(\sigma_1) = M(\sigma_2)) \wedge (\overline{q}_2' \neq F) \wedge (L_{G^R}((x_2, \overline{q}_2), \sigma_2, (x_2', \overline{q}_2'))) \neq \emptyset)$ holds, and 0 otherwise.

The testing automaton T defined in Definition 2 can be used to identify ambiguous states. Following the definition of T, a pair of two states $(x_1, \overline{q}_1)$ and $(x_2, \overline{q}_2)$ of G^R can be reached ambiguously if and only if $(x_1, \overline{q}_1), (x_2, \overline{q}_2) \in Z$.

Example 4: For the system presented in Fig. 1, the testing automaton is shown in Fig. 2. Therefore, $(3, F)$ has two ambiguous nonfaulty states, namely, $(1, 1)$ and $(2, 2)$. This can also be verified through Fig. 2 that $(3, F)$ and $(1, 1)$ can be reached ambiguously by $af(ba)^*$ and $a(ba)^*$, respectively, and $(3, F)$ and $(2, 2)$ can be reached ambiguously by $afb(ab)^*$ and $ab(ab)^*$, respectively.

III. Verification of A-Diagnosability

It has been proven in [19] that the verification of A-Diagnosability is PSPACE-Complete, and therefore a polynomial algorithm for verifying A-Diagnosability likely does not exist. In this letter, we propose a probabilistic test algorithm that requires only polynomial complexity, but at the same time is subject to test error. We start by developing a new necessary and sufficient test condition in this section, followed by the proposed test algorithm in the next section.

Denote a component of G^R as a faulty component if it contains a state whose second coordinate is F, and otherwise, it is denoted as a nonfaulty component. The following theorem provides a new necessary and sufficient condition to test A-Diagnosability, which reduces the verification of A-Diagnosability to that of language equivalence.

Theorem 1: Given a stochastic DES G, deterministic nonfault specification R with generated languages $L = L(G)$ and $K = L(R)$, (G, R) is not A-Diagnosable if and only if there exists a closed faulty SCC C_1 of G^R such that,

- it can be reached ambiguously with another nonfaulty component C_2, i.e., there exists $(x_1, F) \in C_1$ and $(x_1, \overline{q}_2) \in C_2$ such that $((x_1, F), (x_1, \overline{q}_2)) \in Z$, and
- the generated masked language of C_1 starting from (x_1, F) is a subset of the masked language generated by all ambiguous nonfaulty C_2 starting from $(x_2, \overline{q}_2)$, i.e., $M(L(C_1, (x_1, F))) \subseteq \cup_{C_2} M(L(C_2, (x_2, \overline{q}_2)))$.

Proof: When the above condition holds, then there exists $s \in L - K$, after executing which G^R reaches (x_1, F). Therefore, $Pr_{\text{amb}}(s) > 0$. Furthermore, since C_1 is closed and $M(L(C_1, (x_1, F))) \subseteq \cup_{C_2} M(L(C_2, (x_2, \overline{q}_2)))$, all extensions of s are ambiguous with another nonfaulty trace. Therefore, $\forall t \in L\backslash s$, $Pr_{\text{amb}}(st) > 0$. In other words, $\forall n, Pr(t : t \in L\backslash s, |t| \geq n, Pr_{\text{amb}}(st) > 0) = 1$. Therefore, (G, R) is not A-Diagnosable, according to Definition 1.

When the above condition does not hold, then for all faulty traces $s \in L - K$, they either do not lead to an ambiguous SCC, or the generated masked language of C_1 is not a subset of the masked language generated by all ambiguous components C_2. In the former case, $Pr_{\text{amb}}(s) = 0$. In the latter case, let $t_1 \in L\backslash s$ be the shortest extension such that $M(t_1) \notin \cup_{C_2} M(L(C_2, (x_2, \overline{q}_2)))$. Then, $Pr_{\text{amb}}(st_1) = 0$. Denote $Pr(t_1)$ as p_1. Since the conditions in Theorem 1 do not hold, then for all other extensions t of s, either $Pr_{\text{amb}}(st) = 0$ or there exists an extension $t_2 \in L\backslash st$ such that $Pr_{\text{amb}}(stt_2) = 0$. Denote $Pr(t_2)$ as p_2. Let n_k be the length of the kth shortest unambiguous extension of s. Then we have

$$\begin{aligned} &Pr(t : t \in L\backslash s, |t| \geq n_k, Pr_{\text{amb}}(st) > 0) \\ &\leq \prod_{i=1}^{k} Pr(t_i : t_i \in L\backslash st_1 \ldots t_{i-1}, |t_i| = n_i - n_{i-1}, \\ &\qquad Pr_{\text{amb}}(st_1 \ldots t_i) > 0) \\ &\quad \times Pr(t \in L\backslash st_1 \ldots t_k, Pr_{\text{amb}}(st_1 \ldots t_k t) > 0) \\ &\leq \prod_{i=1}^{k} (1 - p_i) \times Pr(t \in L\backslash st_1 \ldots t_k, Pr_{\text{amb}}(st_1 \ldots t_k t) > 0) \\ &\leq \prod_{i=1}^{k} (1 - p_i). \end{aligned}$$

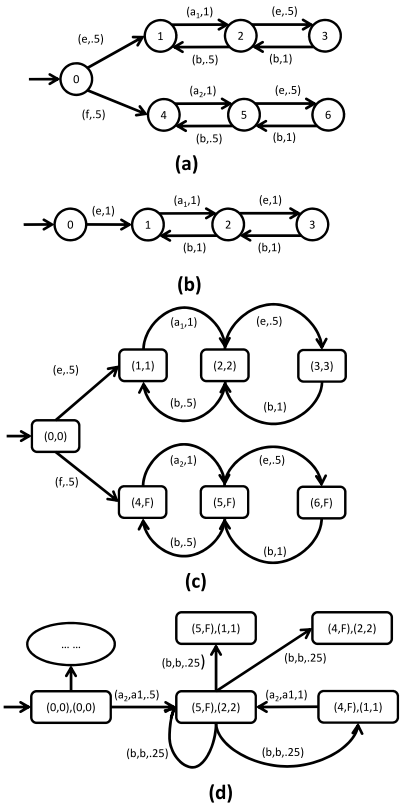

Fig. 3. A non-A-Diagnosable system [19], where $M(f) = M(e) = \epsilon$, $M(a_1) = M(a_2) = a$ and $M(b) = b$; (a) System G, (b) Specification R, (c) Refined system G^R, (d) Testing automaton T.

which approaches 0 when k increases (or equivalently when n_k increases), since $1 - p_i < 1$ for all i. Therefore, for any $\tau > 0$, there should exist $n > 0$, such that $Pr(t : t \in L \backslash s, |t| \geq n, Pr_{\text{amb}}(st) > 0) < \tau$. Note that the above analysis holds for any faulty trace $s \in L - K$. Therefore, (G, R) is A-Diagnosable, according to Definition 1. ■

Example 5: For the system presented in Figs. 1 and 2, C_1 of G^R consisting of $(3, F)$ is the only closed faulty SCC and it can be reached ambiguously with another nonfaulty component C_2 consisting of $(1, 1)$ and $(2, 2)$, as discussed in Example 4. However, the generated masked language of C_1 includes $(aa)^+\{a, b\}^*$ which is unambiguous from all the traces generated by C_2. In other words, $M(L(C_1, (3, F))) \not\subseteq M(L(C_2, (1, 1))) \cup M(L(C_2, (2, 2)))$. Therefore, conditions in Theorem 1 are violated and (G, R) is A-Diagnosable, same as discussed in Example 3.

Example 6: Let's look at another example, which was also studied in [19]. Consider the stochastic plant model G and deterministic nonfault specification generator R as shown in Fig. 3, where f is a fault event and unobservable. The mask function is defined as: $M(f) = M(e) = \epsilon$, $M(a_1) = M(a_2) = a$ and $M(b) = b$. The behaviors after the occurrence of f, as well as e, are identical under the observation mask M, and therefore clearly the system is not A-Diagnosable, according to Definition 1. On the other hand, Theorem 1 can be used to test the A-Diagnosability. As shown in Fig. 3(c), G^R has only one closed faulty SCC C_1 consisting of $(4, F)$, $(5, F)$, and $(6, F)$, which can be reached ambiguously with the nonfaulty component C_2 consisting of $(1, 1)$, $(2, 2)$, and $(3, 3)$. More specifically, $(4, F)$ can be ambiguously reached by $(1, 1)$ and $(2, 2)$, and $(5, F)$ can be ambiguously reached by $(1, 1)$ and $(2, 2)$ as well. Moreover, it is apparent that $M(L(C_1, (x_1, F))) \subseteq M(L(C_2, (x_2, \overline{q}_2)))$ for all ambiguous pair. Therefore, according to Theorem 1, the system is not A-Diagnosable.

Remark 1: Note that the complexity to check the conditions in Theorem 1 is exponential in the number of states in G. Due to the partial observability and nondeterminism of G (and so G^R), checking the condition $M(L(C_1, (x_1, F))) \subseteq \cup_{C_2} M(L(C_2, (x_2, \overline{q}_2)))$ requires an exponential complexity. In fact, according to [19], verification of A-Diagnosability is PSPACE-Complete, and a polynomial verification algorithm likely does not exist.

IV. Probabilistic Test Algorithm

Now we are ready to present the proposed probabilistic test algorithm for checking A-Diagnosability that only requires polynomial complexity. Recall that conditions in Theorem 1 require the set of observations starting from any ambiguous recurrent faulty state to be a subset of that of an ambiguous state in a nonfaulty component. Instead of checking all extensions, the proposed probabilistic test algorithm then randomly selects, for each ambiguous recurrent faulty state, N extensions to check their ambiguity against nonfaulty traces. The complexity required to conduct such a test is then polynomial in the number of system states and events, as well as linear in N. However, since the proposed test only checks N extensions (rather than all extensions), the test results are subject to errors, which will also be analyzed later in this section.

The proposed probabilistic test algorithm is presented in Algorithm 1. For each ambiguous faulty state (x_1, F) that is part of a closed faulty SCC in G^R (Lines 2-5), randomly pick one of its extension t such that $|t| = |X| \times |\overline{Q}|$ (Line 9). Check whether $M(t) \in \cup_{C_2} M(L(C_2, (x_2, \overline{q}_2)))$ by enumerating all C_2 and $(x_2, \overline{q}_2)$ such that $\overline{q}_2 \neq F$ and $((x_1, F), (x_2, \overline{q}_2)) \in Z$ (Lines 10-11). For each $(x_2, \overline{q}_2)$, one will need to see if $M(t) \in M(L(C_2, (x_2, \overline{q}_2)))$, which can be done iteratively by keeping a subcomponent of C_2. See for example [31].

At any time, if it is found that there is $t \in L \backslash s$ such that $M(t) \notin M(L(C_2, (x_2, \overline{q}_2)))$, then the algorithm marks (x_1, F) as diagnosable and moves to the next faulty state. Otherwise, find another random extension t and repeat the above process for N times. If after checking (x_1, F) for N times and the conditions in Theorem 1 have not been found violated, then mark (x_1, F) as not diagnosable. In this case, the algorithm terminates with a decision that (G, R) is not A-Diagnosable (Lines 17-19). After every ambiguous faulty state (x_1, F) that is part of a closed faulty SCC in G^R has been checked for N times, if all of them are marked as diagnosable, then the algorithm terminates with a decision that (G, R) is (possibly) A-Diagnosable (Line 22).

Theorem 2: When (G, R) is indeed not A-Diagnosable, Algorithm 1 can correctly determine it, while when (G, R) is indeed A-Diagnosable, the probability p_d of correctly identifying (G, R) as A-Diagnosable is lower bounded by:

$$p_d = \Pi_s p_s \geq \left\{1 - \left(1 - \frac{1}{|\Sigma|^{|X| \times |\overline{Q}|}}\right)^N\right\}^D \tag{1}$$

TABLE I
PROBABILITY OF ALGORITHM 1

	Algorithm determines `Possibly A-Diagnosable`	Algorithm determines `Not A-Diagnosable`
(G, R) A-Diagnosable	$\geq \left\{1 - \left(1 - \frac{1}{\lvert\Sigma\rvert^{\lvert X\rvert \times \lvert\overline{Q}\rvert}}\right)^N\right\}^D$	$\leq 1 - \left\{1 - \left(1 - \frac{1}{\lvert\Sigma\rvert^{\lvert X\rvert \times \lvert\overline{Q}\rvert}}\right)^N\right\}^D$
(G, R) not A-Diagnosable	0	1

Algorithm 1: Probabilistic Test for A-Diagnosability

Input: G, R, and N
Output: `Possibly A-Diagnosable`, `Not A-Diagnosable`

1 Construct G^R and T;
2 Identify the set of closed faulty SCCs $\mathcal{C}_1$ in G^R that can be ambiguously reached with another nonfaulty SCC;
3 **for** $C_1 \in \mathcal{C}_1$ **do**
4 Identify the set of nonfaulty SCCs $\mathcal{C}_2$ such that C_1 can be ambiguously reached with at least one SCC in $\mathcal{C}_2$;
5 **for** $(x_1, F) \in C_1$ **do**
6 Identify all $(x_2, \overline{q}_2) \in \mathcal{C}_2$ that are ambiguous with (x_1, F);
7 $n \leftarrow 0$;
8 **while** $n < N$ **do**
9 Randomly sample one trace t starting from (x_1, F) with $|t| = |X| \times |\overline{Q}|$;
10 **for** *each* C_2 *and* $(x_2, \overline{q}_2)$ **do**
11 **if** $M(t) \notin M(L(C_2, (x_2, \overline{q}_2)))$ **then**
12 Go to Line 17;
13 **end**
14 **end**
15 $n \leftarrow n + 1$;
16 **end**
17 **if** $n = N$ **then**
18 Terminate with `Not A-Diagnosable`;
19 **end**
20 **end**
21 **end**
22 Terminate with `Possibly A-Diagnosable`;

where D is the number of ambiguous faulty state in a closed faulty SCC in G^R.

Proof: When (G, R) is indeed not A-Diagnosable, then there exists a closed faulty SCC with a faulty state (x_1, F) such that (x_1, F) is ambiguous from a nonfaulty state $(x_2, \overline{q}_2)$ and the generated marked language from (x_1, F) is ambiguous. In this case, the above algorithm will correctly determine it, regardless of the selection of N.

On the other hand, when (G, R) is indeed A-Diagnosable, then for every closed faulty SCC C_1 with a faulty state (x_1, F), there exists a trace $t \in L(C_1, (x_1, F))$ and a nonfaulty component C_2 and $(x_2, \overline{q}_2) \in C_2$ such that $M(t) \notin M(L(C_2, (x_2, \overline{q}_2)))$. Let's assume there are J such traces t. Then the probability of correctly identifying (x_1, F) as diagnosable is:

$$p_s = 1 - \left(1 - \frac{J}{|\Sigma|^{|X| \times |\overline{Q}|}}\right)^N \geq 1 - \left(1 - \frac{1}{|\Sigma|^{|X| \times |\overline{Q}|}}\right)^N \quad (2)$$

Therefore, the probability of correctly identifying (G, R) as A-Diagnosable is:

$$p_d = \Pi_s p_s \geq \left\{1 - \left(1 - \frac{1}{|\Sigma|^{|X| \times |\overline{Q}|}}\right)^N\right\}^D. \quad (3)$$

This completes the proof. ■

The error probabilities are summarized in Table I. As can be seen, if the system is truly not A-Diagnosable, the proposed algorithm can correctly verify it without any error, regardless of N. When the system is truly A-Diagnosable, the probability of test error is then dependent on N. When N increases, the probability of correctly identifying (G, R) as A-Diagnosable also increases, with the price of checking more traces starting from a single recurrent faulty state. Therefore, N serves as a hyperparameter to balance computation and verification accuracy. On the other hand, when the test algorithm outputs `Possibly A-Diagnosable`, then it can be guaranteed that the system is truly A-Diagnosable, while when the test algorithm outputs `Not A-Diagnosable` then the probability of misclassification decreases as N increases.

Example 7: For the system discussed in Example 6 and Fig. 3, Algorithm 1 will correctly determine its non-A-Diagnosability without any error. For the system presented in Figs. 1 and 2, there is only one closed faulty SCC that is ambiguous, so $D = 1$. It is also trivial to see that $J = 2$ since aa and bb are unambiguous. Therefore, the probability of correctly identifying (G, R) as A-Diagnosable is $p_d = 1 - (1 - \frac{2}{2^2})^N = 1 - \frac{1}{2^N}$, which converges to 1 exponentially as N increases. In particular, when $N = 5$, $p_d = 0.96875$ and when $N = 7$ there is over 99% probability that the proposed Algorithm 1 will return correct results.

Lemma 1: The complexity of Algorithm 1 is polynomial in the number of states and events and linear in the hyperparameter N.

Proof: The number of states in G^R is linear in $|X|$ and $|Q|$, and so is the number of ambiguous recurrent faulty states (x_1, F) to be checked at Line 5. In addition, for each (x_1, F), the number of nonfaulty state $(x_2, \overline{q}_2)$ to be checked at Line 10 is also linear in $|X|$ and $|Q|$. Therefore, the condition at Line 11 needs to be checked at most $N \times |X|^2 \times |Q|^2$ times as N extensions t will be randomly sampled. Note that to check if $M(t) \in M(L(C_2, (x_2, \overline{q}_2))$, one does not necessarily have to enumerate all possibility in $L(C_2, (x_2, \overline{q}_2)$, which is exponential in $|X|$ and $|\Sigma|$. Checking if an automaton accepts a certain string can be done in polynomial complexity [32]. Therefore, the total complexity of Algorithm 1 is polynomial in the number of states and events and linear in N. ■

Example 8: Consider a A-Diagnosable system with $|\Sigma| = 3$, $|X| = 3$, $|\overline{Q}| = 2$, $|J| = 3$, and $D = 1$. Fig. 4 plots the probability of correctly identifying the system A-Diagnosability (accuracy) versus the number of extensions

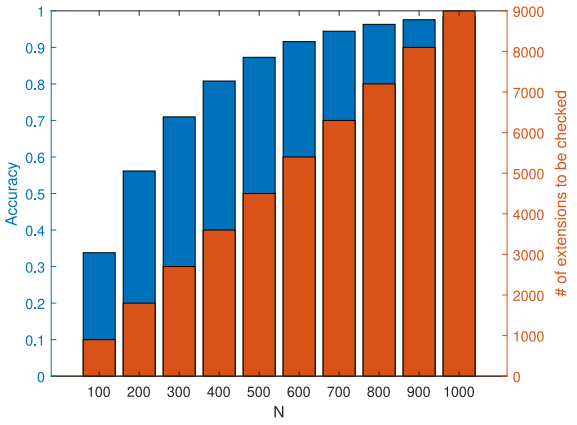

Fig. 4. Comparison of accuracy and computation for different value of N for Example 8.

to be checked (computation). With the increase of N, the required computation grows linearly, while the accuracy grows exponentially.

V. Conclusion

This letter investigates the failure diagnosability of stochastic discrete-event systems (DES). Since the verification of A-Diagnosability has been shown to be PSPACE-Complete, a polynomial testing algorithm likely does not exist. To reduce the verification complexity for large systems, this letter provides a new necessary and sufficient condition for checking A-Diagnosability of stochastic DES, based on which a probabilistic test is also proposed. The proposed probabilistic test randomly samples a subset of faulty trace to be checked and therefore requires a complexity that is polynomial in the number of system states and events. The probability of test error, which is due to the reduction in computation complexity, is also quantified in this letter.

References

[1] Q. Shi and H. Zhang, "Fault diagnosis of an autonomous vehicle with an improved SVM algorithm subject to unbalanced datasets," *IEEE Trans. Ind. Electron.*, vol. 68, no. 7, pp. 6248–6256, Jul. 2021.

[2] X. Kong, Y. Xu, Z. Jiao, D. Dong, X. Yuan, and S. Li, "Fault location technology for power system based on information about the power Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 16, no. 10, pp. 6682–6692, Oct. 2020.

[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] 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.

[6] M. P. Cabasino, A. Giua, S. Lafortune, and C. Seatzu, "A new approach for diagnosability analysis of Petri nets using verifier nets," *IEEE Trans. Autom. Control*, vol. 57, no. 12, pp. 3104–3117, Dec. 2012.

[7] F. Cassez, "The complexity of codiagnosability for discrete event and timed systems," *IEEE Trans. Autom. Control*, vol. 57, no. 7, pp. 1752–1764, Jul. 2012.

[8] W. Dong, X. Yin, and S. Li, "A uniform framework for diagnosis of discrete-event systems with unreliable sensors using linear temporal logic," 2022, *arXiv:2204.13057*.

[9] N. Ran, H. Su, A. Giua, and C. Seatzu, "Codiagnosability analysis of bounded Petri nets," *IEEE Trans. Autom. Control*, vol. 63, no. 4, pp. 1192–1199, Apr. 2018.

[10] R. Su and W. M. Wonham, "Global and local consistencies in distributed fault diagnosis for discrete-event systems," *IEEE Trans. Autom. Control*, vol. 50, no. 12, pp. 1923–1935, Dec. 2005.

[11] X. Yin and S. Lafortune, "Codiagnosability and coobservability under dynamic observations: Transformation and verification," *Automatica*, vol. 61, pp. 241–252, Nov. 2015.

[12] Y. Wang, T.-S. Yoo, and S. Lafortune, "Diagnosis of discrete event systems using decentralized architectures," *Discr. Event Dyn. Syst.*, vol. 17, no. 2, pp. 233–263, 2007.

[13] J. Zaytoon and S. Lafortune, "Overview of fault diagnosis methods for discrete event systems," *Annu. Rev. Control*, vol. 37, no. 2, pp. 308–320, 2013.

[14] E. Athanasopoulou, L. Li, and C. N. Hadjicostis, "Maximum likelihood failure diagnosis in finite state machines under unreliable observations," *IEEE Trans. Autom. Control*, vol. 55, no. 3, pp. 579–593, Mar. 2010.

[15] H. Bazille, E. Fabre, and B. Genest, "Diagnosability degree of stochastic discrete event systems," in *Proc. IEEE 56th Annu. Conf. Decis. Control*, Melbourne VIC, Australia, Dec. 2017, pp. 5726–5731.

[16] J. Chen and R. Kumar, "Failure detection framework for stochastic discrete event systems with guaranteed error bounds," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1542–1553, Jun. 2015.

[17] J. Chen and R. Kumar, "Fault detection of discrete-time stochastic systems subject to temporal logic correctness requirements," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 4, pp. 1369–1379, Oct. 2015.

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

[19] J. Chen, C. Keroglou, C. N. Hadjicostis, and R. Kumar, "Revised test for stochastic diagnosability of discrete-event systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 1, pp. 404–408, Jan. 2018.

[20] X. Yin, Z. Li, L. Zhang, and M. Han, "Distributed state estimation of sensor-network systems subject to Markovian channel switching with application to a chemical process," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 6, pp. 864–874, Jun. 2018.

[21] F. Liu and D. Qiu, "Safe diagnosability of stochastic discrete event systems," *IEEE Trans. Autom. Control*, vol. 53, no. 5, pp. 1291–1296, Jun. 2008.

[22] X. Yin, J. Chen, Z. Li, and S. Li, "Robust fault diagnosis of stochastic discrete event systems," *IEEE Trans. Autom. Control*, vol. 64, no. 10, pp. 4237–4244, Oct. 2019.

[23] F. Liu, D. Qiu, H. Xing, and Z. Fan, "Decentralized diagnosis of stochastic discrete event systems," *IEEE Trans. Autom. Control*, vol. 53, no. 2, pp. 535–546, Mar. 2008.

[24] X. Geng, D. Ouyang, X. Zhao, and S. Hao, "Probabilistic logical approach for testing diagnosability of stochastic discrete event systems," *Eng. Appl. Artif. Intell.*, vol. 53, pp. 53–61, Aug. 2016.

[25] X. Yin and Z. Li, "Decentralized fault prognosis of discrete-event systems using state-estimate-based protocols," *IEEE Trans. Cybern.*, vol. 49, no. 4, pp. 1302–1313, Apr. 2019.

[26] J. Chen and R. Kumar, "Stochastic failure prognosability of discrete event systems," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1570–1581, Jun. 2015.

[27] R. Kumar and V. K. Garg, "Control of stochastic discrete event systems modeled by probabilistic languages," *IEEE Trans. Autom. Control*, vol. 46, no. 4, pp. 593–606, Apr. 2001.

[28] Y. Ji, X. Yin, and S. Lafortune, "Optimal supervisory control with mean payoff objectives and under partial observation," *Automatica*, vol. 123, Jan. 2021, Art. no. 109359.

[29] 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.

[30] J. Chen and R. Kumar, "Polynomial test for stochastic diagnosability of discrete-event systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 4, pp. 969–979, Oct. 2013.

[31] S. Xu and R. Kumar, "Distributed state estimation in discrete event systems," in *Proc. Amer. Control Conf.*, St. Louis, MO, USA, Jul. 2009, pp. 4735–4740.

[32] E. M. Clarke, "Model checking," in *Proc. Found. Softw. Technol. Theor. Comput. Sci. 17th Conf.*, 1997, pp. 54–56.