Diagnosability of Stochastic Discrete-Event Systems

David Thorsley and Demosthenis Teneketzis, *Fellow, IEEE*

***Abstract*—We investigate diagnosability of stochastic discrete-event systems. We define the notions of A- and AA-diagnosability for stochastic automata; these notions are weaker than the corresponding notion of diagnosability for logical automata introduced by Sampath *et al.* Through the construction of a stochastic diagnoser, we determine offline conditions necessary and sufficient to guarantee A-diagnosability and sufficient to guarantee AA-diagnosability. We also show how the stochastic diagnoser can be used for on-line diagnosis of failure events. We illustrate the results through two examples from HVAC systems.**

***Index Terms*—Discrete-event systems, failure detection, fault diagnosis, probabilistic models, stochastic automata.**

I. Introduction

IN THIS paper we investigate the diagnosability of stochastic discrete-event systems (DESs) or stochastic automata. DESs are systems whose evolution is guided by the occurrence of physical events that are separated by regular or irregular intervals of time. Stochastic automata are a more precise formulation of the general DES model, in which a probabilistic structure is appended to the model to estimate the likelihood of specific events occurring. An introduction to the theory of stochastic automata can be found in [2].

The failure diagnosis problem for DES is to detect the occurrence of specific predefined failure events that may not be directly observed by the sensors available to the system. Roughly speaking, a system is considered to be diagnosable if any instance of a failure event can eventually be detected from the observations made on the system.

The problem of diagnosability of logical DES has received a lot of attention in recent years in the contexts of centralized systems [1], [3]–[10], decentralized systems [11], timed systems [12], [13], modeling of systems [14]–[17], and applications [18]–[24]. Diagnosability of stochastic automata was investigated by Lunze and Schröder [25]. The approach to diagnosability of stochastic automata we adopt in this paper is similar to that of [1]; comparisons between our results and those of [1] will be made throughout the rest of this paper.

The main differences between the results of this paper and those on diagnosability of logical DES are the following. Logical DES models cannot distinguish between strings or states that are highly probable and those that are less probable. Therefore, the notions that a state can be observed or a failure can be diagnosed after a finite delay are "all-or-nothing" propositions: One possible system behavior, however improbable, that does not allow the failure to be diagnosed is sufficient to consider a system to be nondiagnosable. In this paper, we present definitions of diagnosability that allow such improbable system behaviors to be disregarded.

The main differences between are results and those of [25] are the following. Our notions of diagnosability are distinct from those of [25]; therefore, the conditions for diagnosability determined in this paper are distinct from those of [25]. Lunze and Schröder correctly demonstrate that, in general, the observer or diagnoser of a stochastic automaton cannot be itself realized by another stochastic automaton, and do not attempt to extend the logical diagnoser approach of [1] to stochastic systems. The "stochastic diagnoser" introduced in this paper inherits the structure of the logical diagnoser of [1], and appends to each transition a matrix that can be used to update the probability distribution on the state estimate. The resulting machine is not a stochastic automaton, but possesses a structure superficially similar to one.

The main contributions of this paper are: 1) the introduction of two notions of diagnosability that appropriately incorporate the stochastic structure of the automaton; these notions of diagnosability are, as expected, less stringent than those of [1]; and 2) the determination of necessary or sufficient conditions to guarantee the aforementioned notions of diagnosability. Preliminary versions of the results in this paper previously appeared in [26].

Because the model under consideration here allows us to formulate the probability distributions of various state estimates and failures, we consider two different definitions of what it means for a failure to be "diagnosed." In the first situation, a failure is not said to be diagnosed until all possible system behaviors consistent with the observations of the system contain at least one instance of the failure event. In the second situation, we merely require that of all the consistent system behaviors, the subset that contains the failure event has a probability above a predetermined threshold. Both notions of diagnosability display long term properties approximating the type of diagnosability proposed in [1]. The conditions necessary and sufficient or sufficient to ensure both types of diagnosability can be expressed in terms of properties of a "diagnoser" that is the stochastic analogue of that in [1].

This paper is organized as follows. Section II introduces the stochastic automaton model under consideration and the concepts and notation required to define diagnosability. Section III introduces new definitions of diagnosability motivated by the probabilistic nature of the automaton. Section IV describes the construction of a stochastic diagnoser used to state conditions that ensure diagnosability. These conditions are presented in

Manuscript received December 23, 2003; revised November 17, 2004. Recommended by Associate Editor H. Wang. This work was supported in part by the National Science Foundation under Grants ECS-0080406 and CCR-0325571, by the Office of Naval Research under Grant N00014-03-1-232, and by a grant from the Xerox University Affairs Committee.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122 USA (e-mail: dthorsle@eecs.umich.edu; teneketzis@eecs.umich.edu).

Digital Object Identifier 10.1109/TAC.2005.844722

0018-9286/$20.00 © 2005 IEEE

Section V and illustrated using examples from HVAC systems in Section VI. The results of the paper are summarized in Section VII.

II. Model

This section describes the formal model of the type of system we will attempt to diagnose and introduces the basic concepts and notation necessary to approach the problem.

The type of system to be diagnosed is a *stochastic automaton*. It is a quadruple

$$G = (\Sigma, \mathcal{X}, p, x_0) \tag{1}$$

where

- Σ is a finite set of events;
- $\mathcal{X}$ is a finite state–space;
- $p(x', e \mid x)$ is a state transition probability defined for $\forall x, x' \in \mathcal{X}, e \in \Sigma$;
- $x_0 \in \mathcal{X}$ is the initial state.

The system evolves through the triggering of events at discrete points in time, forming a sequence, or *string*, of events. The set of all finite strings of positive probability is the prefix-closed language $\mathcal{L}(G)$, which for simplicity will be denoted as L. L is a subset of Σ^*, the Kleene-closure of the event set Σ.

The system is observed through the events transitioning the system from one state to another. The event set Σ is partitioned as $\Sigma = \Sigma_o \dot{\cup} \Sigma_{uo}$, where Σ_o represents the set of observable events and Σ_{uo} represents the set of unobservable events. Observable events are events the occurrence of which is detected by the sensors available to the system; unobservable events are those events that the available sensors cannot detect.

When a string of events occurs in a system, the sequence of observable events is indicated by the *projection* of the string, which is defined in the usual manner [27] as $Pj : \Sigma^* \to \Sigma_o^*$

$$\begin{aligned} Pj(\epsilon) &= \epsilon \\ Pj(\sigma) &= \begin{cases} \sigma, & \text{if } \sigma \in \Sigma_o \\ \epsilon, & \text{if } \sigma \in \Sigma_{uo} \end{cases} \\ Pj(s\sigma) &= Pj(s)Pj(\sigma), \qquad \text{for } s \in \Sigma^*, \sigma \in \Sigma \end{aligned} \tag{2}$$

where ϵ denotes the empty string. The inverse projection of a string of observable events s_o with respect to a language L is given by

$$Pj_L^{-1}(s_o) = \{s \in L : Pj(s) = s_o\}. \tag{3}$$

We define a set of failure events $\Sigma_f \subseteq \Sigma$. The objective of the diagnosis problem under consideration is to determine the likelihood of the occurrence of these failure events when only the events in Σ_o are observed. We can assume, without loss of generality, that $\Sigma_f \subseteq \Sigma_{uo}$, because it is a trivial problem to determine when an observable failure has occurred.

To facilitate the diagnosis problem, the set of failure events is partitioned into a set of failure types

$$\Sigma_f = \Sigma_{f_1} \cup \cdots \cup \Sigma_{f_m}. \tag{4}$$

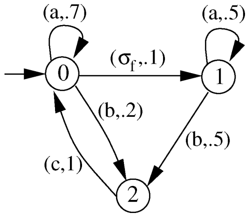

Fig. 1. Stochastic automaton. $\mathcal{X} = \{0, 1, 2\}$, $\Sigma = \{a, b, c, \sigma_f\}$, $\Sigma_{uo} = \Sigma_f = \{\sigma_f\}$, and $x_0 = 0$.

If a failure event $\sigma_f \in \Sigma_{f_i}$ occurs, we will say that "a failure of type F_i has occurred."

The probability transition function $p(x', e \mid x)$ defines the probability that a certain event e will occur and transition the state of the machine from a given state x to the specified state x'. For example, $p(z, a \mid y) = 0.7$ states that, if the system is in state y, with probability 0.7 the event a will occur and transition the state of the system to z. We will assume, for the sake of simplicity, that $p(x', e \mid x) > 0$ for at most one $x' \in \mathcal{X}$. The results that follow hold without this assumption.

Under this assumption, the probability transition function can be used to define the *partial transition function*, which is defined as $\delta : \mathcal{X} \times \Sigma \to \mathcal{X}$ where

$$\delta(x, e) = y \Rightarrow p(y, e \mid x) > 0. \tag{5}$$

If for some $x \in \mathcal{X}$ and $e \in \Sigma$, there does not exist $y \in \mathcal{X}$ such that $p(y, e \mid x) > 0$, then $\delta(x, e)$ is undefined.

This relationship demonstrates that the stochastic model is a more specific model than the logical finite state machine model discussed in [1]. In the logical automaton model, the partial transition function is defined as part of the specification of the system, but here it is derived from the state transition probabilities.

The partial transition function can be extended to strings of events as follows:

$$\begin{aligned} \delta(x, \epsilon) &= x & (6) \\ \delta(x, se) &= \delta(\delta(x, s), e)). & (7) \end{aligned}$$

Fig. 1 provides a pictorial representation of a stochastic automaton. The set of states is $\mathcal{X} = \{0, 1, 2\}$, and the initial state $x_0 = 0$ is denoted by an unconnected transition. The set of events is $\Sigma = \{a, b, c, \sigma_f\}$, which is partitioned into $\Sigma_o = \{a, b, c\}$ and $\Sigma_{uo} = \Sigma_f = \{\sigma_f\}$. A transition arc is drawn between two states if the probability of that transition occurring is greater than zero.

In order to facilitate the solution to the diagnosis problem, we make two assumptions about the stochastic automaton G:

(A1): The language L generated by G is live. That is to say, for every state in $\mathcal{X}$, the probability of a transition occurring from that state is one or, equivalently, for $\forall x \in \mathcal{X}$

$$\sum_{x' \in \mathcal{X}} \sum_{e \in \Sigma} p(x', e \mid x) = 1. \tag{8}$$

(A2): The generator G does not have any cycle of unobservable events or, equivalently

$$\exists n_0 \in \mathbb{N} \text{ such that } \forall ust \in L, s \in \Sigma_{uo}^* \Rightarrow \|s\| \leq n_0.$$

Together these two assumptions force observable events to occur with regularity. Assumption (A1) requires that transitions will continue to occur regardless of the state of the system, and (A2) requires that after at most a finite delay, one of these transitions will be an observable event.

The liveness assumption (A1) also forces all states in G to satisfy the Markov property, allowing the use of techniques of Markov chain analysis in subsequent sections of this paper.

A. Discrete-Event Notation

The symbol $\overline{s}$ will be used to denote the prefix-closure of a string $s \in \Sigma^*$. The postlanguage L/s is the set of possible continuations of a string s, i.e.,

$$\frac{L}{s} = \{t \in \Sigma^* \mid st \in L\}. \tag{9}$$

When defining diagnosability, it will be important to consider strings that end in a failure event of a specific type. Let the final event of a string s be denoted by s_f. Define

$$\Psi(\Sigma_{f_i}) = \{s \in L : s_f \in \Sigma_{f_i}\}. \tag{10}$$

Let $L(G, x)$ represent the set of all strings that originate from the state x in the state–space of G. Define

$$L_o(G, x) = \{s \in L(G, x) : s = u\sigma, u \in \Sigma_{uo}^*, \sigma \in \Sigma_o\} \tag{11}$$

$$L_\sigma(G, x) = \{s \in L_o(G, x) : s_f = \sigma\}. \tag{12}$$

Convention: In the examples of stochastic automata in this paper, observable events will be marked using lowercase Roman letters, while unobservable events that are not failure events will be denoted by lowercase Greek letters. Failures will be denoted as σ_{f_i}.

B. Probabilistic Notation

From our assumption that $p(x', e \mid x) > 0$ for only one x' for each pair (x, e), we can write $p(x', e \mid x) = p(e \mid x)$. Therefore, the probability of an event $e \in \Sigma$ being the next event given the system is in state x is given by

$$Pr(e \mid x) = p(e \mid x). \tag{13}$$

If we wish to find the probability of a particular string being the true future system behavior given the system is state x, we can calculate this recursively as

$$Pr(es \mid x) = p(e \mid x)p(s \mid \delta(x, e)). \tag{14}$$

Because our tests for diagnosability can only be based on observable events, it will be important to determine the probability that $e_o \in \Sigma_o$ will be the next observable event given the system is in state x. This can be calculated as

$$Pr(e_o \mid x) = \sum_{s \in L_{e_o}(x)} Pr(s \mid x). \tag{15}$$

If our state observation is incomplete, we will need to determine the probability being in a state x, given that we have observed the string s_o. This probability is

$$Pr(x \mid s_o) = \frac{\sum\limits_{s \in Pj_L^{-1}(s_o) : \delta(x_o, s) = x} Pr(s \mid x_0)}{\sum\limits_{s \in Pj_L^{-1}(s_o)} Pr(s \mid x_0)}. \tag{16}$$

By combining (15) and (16), we can determine the probability of the next observable event being e_o, given that the string of observed events to date is s_o

$$Pr(e_o \mid s_o) = \sum_{x \in \delta(x_0, s) \text{ for some } s \in Pj_L^{-1}(s_o)} Pr(e_o \mid x) Pr(x \mid s_o). \tag{17}$$

Finally, if we are in a state x, the probability that after the next event e_o, the system has been transitioned to the state x' is given by

$$Pr(x', e_o \mid x) = \sum_{s \in L_{e_o}(x) : \delta(x, s) = x'} Pr(s \mid x). \tag{18}$$

III. Approaches to Defining Diagnosability

The objective of the diagnosis problem is to detect the occurrence of an unobservable failure in the system, based on the information available from the record of observed events. As a starting point for motivating the discussion of diagnosability of stochastic automata, we present a standard definition of logical diagnosability defined in [1].

Definition 1: (Logical Diagnosability): A live, prefix-closed language L is F_i-diagnosable with respect to a projection Pj if

$$(\exists n_i \in \mathbb{N})[\forall s \in \Psi(\Sigma_{f_i})] \left(\forall t \in \frac{L}{s}\right) [\|t\| \geq n_1 \Rightarrow D] \tag{19}$$

where the diagnosability condition function $D : \Sigma^* \rightarrow \{0, 1\}$ is given by

$$D(st) = \begin{cases} 1, & \text{if } \omega \in Pj_L^{-1}[Pj(st)] \Rightarrow \Sigma_{f_i} \in \omega \\ 0, & \text{otherwise} \end{cases}. \tag{20}$$

Fig. 2 shows an example of a system that is F-diagnosable. If the event σ_f occurs, the next observable event will be either a or b. If b is observed, then we know that the only possible system behavior consists with our observation of b is $\sigma_f b$, and the failure will be diagnosed. On the other hand, if a is observed, it will necessarily be followed by c. The only behavior consistent with the string of observations ac is $\sigma_f ac$, so once again the failure will be diagnosed. Regardless of whether the first observed event is a or b, in this example we will not have to wait

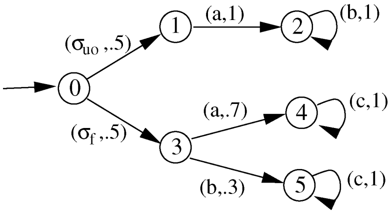

Fig. 2. System that is logically diagnosable.

for more than two events after the failure to determine that the failure has indeed occurred.

This definition of diagnosability was developed for logical automata models and the necessary and sufficient conditions for this type of diagnosability are stated in [1]. This definition, therefore, makes no use of the probabilistic information that the stochastic model under consideration in this paper contains. We now present weaker definitions of diagnosability that take into account the stochastic structure of the model.

A. A-Diagnosability

Consider the system in Fig. 3, and suppose that the behavior of the system is the string $s = \sigma_f$. Clearly, $s \in \Psi(\Sigma_{f_i})$. The postlanguage of s is given by $L/s = a^*b^*$, meaning that it consists of an arbitrary number of a's followed by an arbitrary number of b's.

Let $n \in \mathbb{N}$. Let $t \in L/s$ such that $||t|| = n$. Then, t is of the form $a^{n-k}b^k$, $0 \le k \le n$.

Suppose $k > 0$. Then, $Pj_L^{-1}[Pj(st)] = \sigma_f a^{n-k}b^k$. The failure is therefore diagnosed, as every string in the projection contains the failure event F.

Now, suppose $k = 0$, that is to say, $t = a^n$. Then, $Pj_L^{-1}[Pj(st)] = \{\sigma_f a^n, \mu a^n\}$. The logical diagnosability condition is not satisfied, as $\sigma_f \notin \mu a^n$. Since the string a^n is part of the postlanguage L/s for an arbitrarily large n, there is potentially an infinite delay before the failure can be diagnosed. Therefore, the system is not logically diagnosable.

The string that does not allow us to declare the systems to be diagnosable is $a^n \in L/s$, the only continuation after the failure event along which the failure cannot be diagnosed. Because we have appended probabilities to the system model considered in [1], we can now consider the probability of the string a^n being the actual behavior of the system. At each moment, the probability of a occurring is 0.9, so $Pr(a^n) = (0.9)^n$. As n increases, the probability of the set of strings that do not allow diagnosis approaches zero.

Although in this system we can never guarantee that the failure will be diagnosed after a finite delay, the probability of a string of events that allows diagnosis becomes greater as we observe more events after the failure event.

This observation is the motivation for this weaker definition of diagnosability, which is created by making (19) less stringent.

Definition 2: (A-Diagnosability): A live, prefix-closed language L is F_i-A-diagnosable with respect to a projection Pj and a set of transition probabilities p if

$$(\forall \epsilon > 0)(\exists N \in \mathbb{N})(\forall s \in \Psi(\Sigma_{f_i}) \wedge n \geq N)$$
$$\left\{ Pr\left(t : D(st) = 0 \mid t \in \frac{L}{s} \wedge ||t|| = n \right) < \epsilon \right\} \quad (21)$$

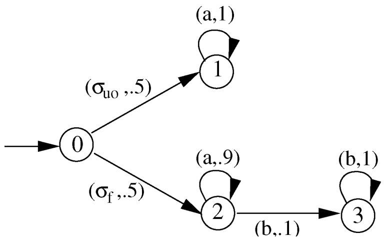

Fig. 3. System that is A-diagnosable but not logically diagnosable.

where the diagnosability condition function D is as in (20)

$$D(st) = \begin{cases} 1, & \text{if } \omega \in Pj_L^{-1}[Pj(st)] \Rightarrow \Sigma_{f_i} \in \omega \\ 0, & \text{otherwise} \end{cases}. \quad (22)$$

The system in Fig. 3 is F_i-A-diagnosable. The only string in the postlanguage L/s that does not allow diagnosis of the failure F_i is a^n, a string whose probability of occurring approaches zero as n becomes arbitrarily large. This indicates that, after a failure occurs, we can let the probability of diagnosing the failure after a finite delay become arbitrarily high by selecting a value n such that $Pr(a^n)$ is sufficiently small.

B. AA-Diagnosability

The definition of stochastic diagnosability presented in Section III-B is not the only way in which the diagnosability conditions in Definition 1 can be weakened using a probabilistic model. Consider the system shown in Fig. 4. Let $s = a\sigma_f$. The postlanguage L/s is given by $\overline{(ba + ba\sigma_f)^*}$. Let $t \in L/s$ and let $m \in \mathcal{N}$ be an integer such that t contains $2m$ observable events. Then, $Pj(st) = a(ba)^m$, and $Pj_L^{-1}[Pj(st)] = (ab + a\sigma_f b)^m a$.

The string $(ab)^m a$ is part of the set $Pj_L^{-1}[Pj(st)]$, regardless of the length of the continuation t. Similarly, if t has $2m - 1$ observable events, then $(ab)^m \in Pj_L^{-1}[Pj(st)]$.

Because $\sigma_f \notin (ab)^m$ and $\sigma_f \notin (ab)^m a$, the diagnosability condition function $D(st)$ is equal to zero for all continuations t of the string s. Therefore this system is neither diagnosable nor A-diagnosable, as we can never say for any continuation t that all possible true system behaviors consistent with the observed behavior contain the failure event σ_f.

The problematic string is $(ab)^m$. However, just as when the condition for A-diagnosability was developed, we can consider the probability of the string $(ab)^m$ which does not contain the failure event σ_f. The probability of this string is $(1)(0.9)(1)(0.9) \cdots (1)(0.9) = (0.9)^m$.

The probability of the string that does not contain a failure approaches zero as the number of events observed becomes large. Therefore, although we cannot assure that a correct diagnosis is made, we can force the probability of the failure event being included in the actual behavior to be arbitrarily close to one by waiting for a sufficiently long, yet finite, amount of observations.

This observation allows us introduce a third notion of diagnosability that is weaker than both logical diagnosability and A-diagnosability:

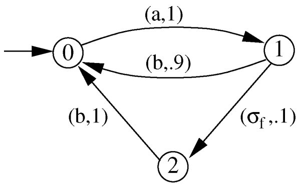

Fig. 4. System that is AA-diagnosable but not A-diagnosable.

Definition 3: (AA-Diagnosability): A live, prefix-closed language L is F_i-AA-diagnosable with respect to a projection Pj and a transition probability function p if

$$(\forall\epsilon > 0 \wedge \forall\alpha < 1)(\exists N \in \mathbb{N})(\forall s \in \Psi(\Sigma_{f_i}) \wedge n \geq N)$$
$$\left\{Pr\left(t : D_\alpha(st) = 0 \mid t \in \frac{L}{s} \wedge \|t\| = n\right) < \epsilon\right\} \quad (23)$$

where the diagnosability condition function D_α is

$$D_\alpha(st)=\begin{cases} 1, & \text{if } Pr(\omega : \Sigma_{f_i} \in \omega \mid \omega \in Pj_L^{-1}[Pj(st)]) > \alpha \\ 0, & \text{otherwise} \end{cases}. \quad (24)$$

The system in Fig. 4 is F_i-AA-diagnosable, because as $\|t\|$ becomes large, the probability that the system behavior that does not contain a failure approaches zero.

Roughly speaking, the difference between logical diagnosability and AA-diagnosability can be described as the difference between a "sure" convergence and an "almost sure" convergence. For an automaton to be logically diagnosable, it is required that all strings of a certain length allow the system to be diagnosed, and a system is not considered to be diagnosed until all strings consisted with the observed behavior contain the failure event. In AA-diagnosability, we require that "almost all" strings of a certain length will diagnose the failure, and we consider a failure to be diagnosed if "almost all" strings consistent with the observed behavior contain the failure event.

A-diagnosability can be interpreted as a condition halfway between logical diagnosability and AA-diagnosability, as a system is A-diagnosable if almost all strings of a certain length diagnose the failure (as in AA-diagnosability) but we still require that every string consistent with the observed behavior must contain a failure before we diagnose the system (as in logical diagnosability).

In our opinion, the notions of A- and AA-diagnosability present an intuitive approach to defining diagnosability of stochastic systems. This is because we are interested in the behavior of stochastic systems only along strings of nonzero probability; A- and AA-diagnosability are concerned with the behavior of a stochastic automaton along only these strings.

The definitions of A- and AA-diagnosability place conditions on the limiting behavior of the system as the length of a continuation following a failure grows without bound. To test for these limiting properties, we will need to derive offline conditions that are necessary and sufficient to confirm A- and AA-diagnosability.

On the other hand, if we wish to determine if the diagnosability condition $D_\alpha(st) = 1$ is satisfied for a specific α, we can do this by observing the behavior of the system online. Given an observed string s_o, we can calculate the probability that a failure of type F_i has occurred using a machine called the "stochastic diagnoser." If the probability of failure is greater than α, we can declare that a failure has occurred. The stochastic diagnoser can also be used as a tool to help accomplish the primary goal of this paper, which is to determine conditions necessary and sufficient or sufficient to ensure A- and AA-diagnosability.

IV. Stochastic Diagnoser

The stochastic diagnoser is a machine constructed from the stochastic automaton G that can be used either online or offline to describe the behavior of the system. Its construction is based on that of the logical diagnoser of [1].

Offline, the stochastic diagnoser can be used to formulate necessary and/or sufficient conditions for A- and AA-diagnosability.

Online, the stochastic diagnoser is used to determine three pieces of information. First, it estimates the current state of the system after the occurrence of each observable event; secondly, it determines which failure events may have occurred for any such estimate of the system state; and lastly, it calculates the probability of each component of the state estimate.

A state of the logical diagnoser contains the first two of these three pieces of information. In general, a logical diagnoser is a finite state machine since there are a finite number of possible state estimates and a finite number of possible failures. However, in a given stochastic diagnoser there may be infinitely many probability mass functions associated with one logical diagnoser state, and thus a stochastic automaton cannot in general be diagnosed using a finite state machine. This observation is in agreement with the results of [25].

The stochastic diagnoser is thus a compact representation of an infinite state machine. It is useful to think of the information state (see [28]) of the stochastic diagnoser consisting of a "logical element" containing the state estimate and failure information, and a "stochastic element" containing the probabilities of each component of the logical part of the state.

In this section, we describe the construction of the stochastic diagnoser and demonstrate how it can be used to determine the probabilities of failure events online. We also show how the stochastic diagnoser can be described as a Markov chain, and review results from Markov chain theory that will be used to derive conditions for A- and AA-diagnosability in Section V.

A. Construction of the Stochastic Diagnoser

In order to construct the stochastic diagnoser, we need to first define a set of failure labels $\Delta_f = \{F_1, F_2, \ldots F_m\}$ where m is the number of different failure types in the system. The set of possible failure labels is defined as

$$\Delta = \{N\} \cup 2^{\{\Delta_f\}} \quad (25)$$

where $2^{\{\Delta_f\}}$ denotes the power set of Δ_f. The $\{N\}$ label should be interpreted as representing the "normal" behavior of the system, while a label of the form $\{F_i, F_j\}$ should be interpreted to mean that "at least one failure of type i and at least one failure of type j have occurred."

The set of observable states of the stochastic automaton is defined as

$$x \in X_o \Rightarrow (x = x_0) \vee (\exists s \in L : \delta(x_0, s) = x \wedge s_f \in \Sigma_o). \tag{26}$$

The set of possible "logical elements" of the stochastic diagnoser states is defined as follows:

$$Q_o = 2^{X_o \times \Delta}. \tag{27}$$

Each logical element consists of a subset of the observable states of the original system with failure labels attached.

The stochastic diagnoser for a stochastic finite-state machine G is the machine

$$G_d = (Q_d, \Sigma_o, \delta_d, q_0, \Phi, \phi_0) \tag{28}$$

where

- Q_d is the set of logical elements;
- Σ_o is the set of observable events;
- δ_d is the transition function of the diagnoser (to be defined later);
- $q_0 \in Q_d$ is the initial logical element, defined as $\{(x_0, \{N\})\}$;
- Φ is the set of probability transition matrices (to be defined later);
- ϕ_0 is the initial probability mass function on q_0.

As an illustration, the stochastic diagnoser for the system of Fig. 1 is presented in Fig. 5.

The set of logical elements, Q_d, is the subset of Q_o that is reachable from q_0 under δ_d. A logical element $q_d \in Q_d$ is a set of the form

$$q_d = \{(x_1, l_1), \ldots, (x_n, l_n)\}$$

where $x_i \in X_o$ and $l_i \in \Delta$. A pair (x_1, l_1) in a logical element of a diagnoser state is called a *component*. The set of components of the diagnoser is defined as the set of all triples (q, x, l) such that $q \in Q_d$, $x \in X_o$, $l \in \Delta$, and $(x, l) \in q$. The number of components in a logical element q_d will be denoted by $||q_d||$.

In order to construct the probability transition matrices Φ, we will need to impose an order on the set of components in each logical element $q \in Q_d$. This order can be chosen arbitrarily. By convention, the ith component of a state q will be denoted by $c_{q,i}$.

Each logical element of the diagnoser consists of the set of components that are possible true states consistent with the observed system behavior. If the component (x, l) is part of q, it means that for every sequence of observed events that transitions the diagnoser to q, there exists at least one string s in the inverse projection of that sequence such that s transitions the stochastic automaton to the state x and failures of all types included in the label l are included in s. The properties of components of the diagnoser will be essential to providing conditions for A- and AA-diagnosability in Section V.

In order to define δ_d, the transition function of the diagnoser, we must first define how the labels change from one logical element to another. Define the label propagation function $LP : X_o \times \Delta \times \Sigma_o^* \to \Delta$ as

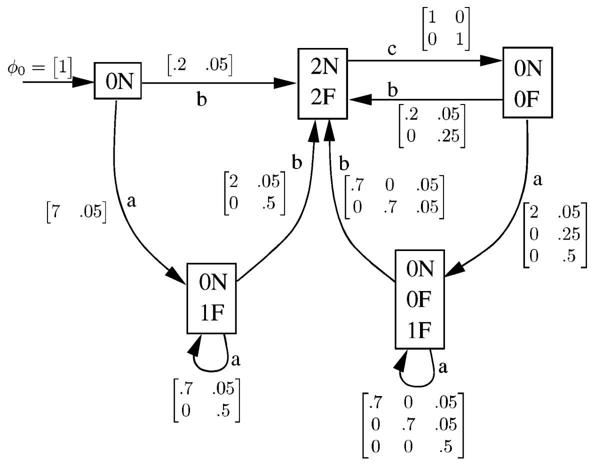
$\phi_0 = [1]$
0N
$[.2 \quad .05]$
b
2N
2F
c
$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
0N
0F
b
$\begin{bmatrix} .2 & .05 \\ 0 & .25 \end{bmatrix}$
$[7 \quad .05]$
a
$\begin{bmatrix} 2 & .05 \\ 0 & .5 \end{bmatrix}$
b
b
$\begin{bmatrix} .7 & 0 & .05 \\ 0 & .7 & .05 \end{bmatrix}$
a
$\begin{bmatrix} 2 & .05 \\ 0 & .25 \\ 0 & .5 \end{bmatrix}$
0N
1F
a
$\begin{bmatrix} .7 & .05 \\ 0 & .5 \end{bmatrix}$
0N
0F
1F
a
$\begin{bmatrix} .7 & 0 & .05 \\ 0 & .7 & .05 \\ 0 & 0 & .5 \end{bmatrix}$

Fig. 5. Stochastic diagnoser of the system in Fig. 1.

$$LP(x, l, s) = \begin{cases} \{N\}, & \text{if } l = \{N\} \wedge \forall i [\Sigma_{f_i} \notin s] \\ \{F_i : F_i \in l \vee \Sigma_{f_i} \in s\}, & \text{otherwise} \end{cases}. \tag{29}$$

Using the label propagation function, we can define the transition function of the diagnoser as

$$\delta_d(q, \sigma) = \bigcup_{(x,l) \in q} \bigcup_{s \in L_\sigma(G,x)} \{(\delta(x, s), LP(x, l, s))\}. \tag{30}$$

The function LP shows that a label F_i is added whenever the true behavior of the system contains an event $\sigma_f \in \Sigma_{f_i}$. Once this label is appended, it cannot be removed regardless of whether or not an event in Σ_{f_i} occurs or not in the system behavior following the label.

From the transition functions of the original stochastic automaton and the stochastic diagnoser, we can define the *component transition function* δ_{comp}, which determines the component of the diagnoser that is the true state of the original system, given that the true behavior of the system is $s \in \Sigma^*$ and $\sigma_f \in \Sigma_o$. Given $q \in Q_d$, $(x, l) \in q$, and s

$$\delta_{\text{comp}}(q, x, l, s) = (\delta_d(q, Pj(s)), \delta(x, s), LP(x, l, s)). \tag{31}$$

The quadruple $(Q_d, \Sigma_o, \delta_d, q_0)$ that has been defined above is equivalent to the diagnoser presented in [1], with the modification that the "ambiguous" label has been removed from the set of possible labels. This quadruple is used to provide estimates of the state and information on the possible failure events. This is the "discrete-event" part of the stochastic diagnoser, which is used to determine the logical element of the diagnoser state. In order to derive the probabilities of each component in the logical element of an information state, we now append a probabilistic structure to make the diagnoser "stochastic."

We define a set of probability transition matrices as $\Phi : Q_d \times \Sigma_o \rightarrow \mathcal{M}_{[0,1]}$

$$\Phi_{ij}(q,\sigma_o) = \sum_{s\in L_{\sigma_o}(G,x_i):(\delta(x_i,s),LP(x_i,l_i,s))=(x_j,l_j)} Pr(s) \tag{32}$$

$$= Pr(c_{\delta_d(q,\sigma_o),j}, \sigma_o \mid c_{q,i}) \tag{33}$$

where the range $\mathcal{M}_{[0,1]}$ represents the set of finite-dimensional matrices whose values are contained in the interval [0, 1]. The size of the matrix outputted by $\Phi(q,\sigma)$ is $\|q\| \times \|\delta_d(q,\sigma_o)\|$. So, for example, if an event takes the diagnoser from a logical element with m components to a logical element with n components, the size of the matrix associated with that event will be $m \times n$. The initial probability vector of the system corresponds to the probability mass function of the initial logical element. Since the only component of the initial logical element is, by construction, $(x_0, \{N\})$, we define $\phi_0 = [1]$

$$\begin{aligned}
&\phi(s_oe_o)\\
&= [\,Pr(c_{j,1} \mid s_oe_o) \quad \cdots \quad Pr(c_{j,m} \mid s_oe_o)\,] && (36)\\
&= \frac{1}{Pr(e_o \mid s_o)}\Big[Pr(c_{j,1} \mid s_oe_o)Pr(e_o \mid s_o)\\
&\quad \cdots Pr(c_{j,m} \mid s_oe_o)Pr(e_o \mid s_o)\Big] && (37)\\
&= \frac{1}{Pr(e_o \mid s_o)}\,[\,Pr(c_{j,1}, e_o \mid s_o) \quad \cdots \quad Pr(c_{j,m}, e_o \mid s_o)\,] && (38)\\
&= \frac{1}{Pr(e_o \mid s_o)}\Big[\sum_{k=1}^{n} Pr(c_{j,1}, e_o \mid c_{i,k}, s_o)Pr(c_{i,k} \mid s_o)\\
&\quad \cdots \sum_{k=1}^{n} Pr(c_{j,m}, e_o \mid c_{i,k}, s_o)Pr(c_{i,k} \mid s_o)\Big] && (39)\\
&= \frac{1}{Pr(e_o \mid s_o)}\,[\,Pr(c_{i,1} \mid s_o) \ldots Pr(c_{i,n} \mid s_o)\,]\\
&\quad \times \begin{bmatrix} Pr(c_{j,1}, e_o \mid c_{i,1}, s_o) & \cdots & Pr(c_{j,m}, e_o \mid c_{i,1}, s_o)\\ \vdots & \ddots & \vdots \\ Pr(c_{j,1}, e_o \mid c_{i,n}, s_o) & \cdots & Pr(c_{j,m}, e_o \mid c_{i,n}, s_o)\end{bmatrix} && (40)\\
&= \frac{\phi(s_o)\Phi(q_i, e_o)}{Pr(e_o \mid s_o)}. && (41)
\end{aligned}$$

In general, the number of information states of a stochastic diagnoser is infinite because there may be a unique ϕ for each sequence of observable events in the system. Since there may be infinitely many unique sequences of observable events, there may be infinitely many reachable probability vectors. The set of matrices Φ are a compact representation of the calculations that are needed to compute the reachable probability vectors along any observed behavior of the stochastic automaton.

Given that the information state of a diagnoser is (q,ϕ), where q is an order set $\{c_1, c_2, \ldots, c_n\}$ and ϕ is a vector $[\phi_1, \phi_2, \ldots, \phi_n]$, we conclude that $Pr(c_i) = \phi_i, i = 1, \ldots, n$. The following theorem makes clear the procedure of calculating the probability vector ϕ from the set of matrices Φ.

Theorem 1: The state probability vector $\phi(s_oe_o)$ can be calculated recursively as follows:

$$\phi(\epsilon) = [1] \tag{42}$$

$$\phi(s_oe_o) = \frac{\phi(s_o)\Phi(q_i, e_o)}{Pr(e_o \mid s_o)}. \tag{43}$$

Proof: If the observed string is the empty string, then, by its construction, the stochastic diagnoser is in the initial logical element and $\phi(\epsilon) = \phi_0 = [1]$. Now, suppose there exists an observable event e_o that transitions the system from a logical element $q_i \in Q_d$ where $\|q_i\| = n$ to $q_j \in Q_d$ where $\|q_j\| = m$. The recursive equation for the probability vector is given by the derivation (36)–(41) ■

The dependence on the observed string s_o in (40) does not affect the values of the terms of the matrix since these probabilities satisfy the Markov property (Assumption A1).

This result gives us a method to perform on-line diagnosis by calculating the probability vector from the matrices in the stochastic diagnoser. Suppose the observed behavior of the system is $s_o = e_1e_2, \ldots, e_n$, and the sequence of observed logical elements is $(q_1, q_2, \ldots, q_n)$. Then, the unnormalized probability vector is given by $\phi_{un}(s_o) = \phi_o\Phi(q_1, e_1)\Phi(q_2, e_2), \ldots, \Phi(q_n, e_n)$. To find the normalized probability, we need to divide this vector by $Pr(e_o \mid s_o)$, which is simply the sum of the terms of $\phi_{un}(s_o)$.

To perform online diagnosis of failures using the stochastic diagnoser, select a threshold β such that $0 < \beta < 1$. Suppose we observe online the string of events s_o, and let $s \in Pj_L^{-1}(s_o)$. We say that a failure has been diagnosed online if $D_\beta(s) = 1$ or, equivalently, if $Pr(F \mid s_o) > \beta$.

Example 1: Consider the diagnoser in Fig. 5, and suppose the observed behavior of the system is $s_o = aabcaa$. Then, the probability vector is given by

$$\begin{aligned}
\phi_{un}(s_o) &= [1]\,[\,.7 \quad .05\,]\begin{bmatrix} .7 & .05 \\ 0 & .5 \end{bmatrix}\begin{bmatrix} .2 & .05 \\ 0 & .5 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\\
&\quad \begin{bmatrix} .7 & 0 & .05 \\ 0 & .7 & .05 \end{bmatrix}\begin{bmatrix} .7 & 0 & .05 \\ 0 & .7 & .05 \\ 0 & 0 & .5 \end{bmatrix}\\
&= [\,0.048\,02 \quad 0.026\,705 \quad 0.009\,15\,]\\
\phi(s_o) &= [\,0.5725 \quad 0.3184 \quad 0.1091\,].
\end{aligned} \tag{44}$$

The logical element reached after observing s_o is $\{(0,N), (0,F), (1,F)\}$. Therefore, the information state of the stochastic diagnoser is

$$(q,\phi) = (\{(0,N), (0,F), (1,F)\}, [0.5725 \quad 0.3184 \quad 0.1091]).$$

From this information state, we can conclude that

$$\begin{aligned}
Pr((0,N) \mid s_o) &= 0.5725\\
Pr((0,F) \mid s_o) &= 0.3184\\
Pr((1,F) \mid s_o) &= 0.1091.
\end{aligned}$$

Therefore

$$Pr(F \mid s_o) = Pr((0,F) \mid s_o) + Pr((1,F) \mid s_o) = 0.4275.$$

If we had set a failure threshold of, say, $\beta = 0.4$, we would now declare that a failure has occurred and take appropriate steps to repair the system. If we had set a higher threshold ($\beta > 0.43$), we would continue to observe the system until such time as a string s_o is observed where $Pr(F \mid s_o) > \beta$.

B. Embedded Markov Chain of the Stochastic Diagnoser

Despite the fact that the stochastic diagnoser has matrices associated with each transition instead of simple probabilities, embedded within a diagnoser is a Markov chain whose states are the components of each state estimate in the diagnoser. The existence of this embedded chain will allow us to use techniques of Markov chain analysis to derive conditions for A- and AA-diagnosability in Section V.

To construct the embedded Markov chain, first define $\Omega : Q_d \times Q_d \to \mathcal{M}_{[0,1]}$ as

$$\Omega(q_i, q_j) = \sum_{e_o \in \Sigma_o : \delta(q_i, e_o) = q_j} \Phi(q_i, e_o). \tag{45}$$

$\Omega(q_i, q_j)$ is simply the sum of all the matrices associated with the transitions from q_i to q_j. If there are no transitions from q_i to q_j, $\Omega(q_i, q_j)$ is a matrix of zeros of size $\|q_i\| \times \|q_j\|$.

We can construct the embedded Markov transition matrix from $\Omega(q_i, q_j)$ as follows.

Theorem 2: Let G_d be a stochastic diagnoser. Then, the matrix $\Pi(G_d)$ is a Markov transition matrix, where $\Pi(G_d)$ is defined as

$$\Pi(G_d) = \begin{bmatrix} \Omega(q_1, q_1) & \dots & \Omega(q_1, q_n) \\ \vdots & \ddots & \vdots \\ \Omega(q_n, q_1) & \dots & \Omega(q_n, q_n) \end{bmatrix}. \tag{46}$$

Proof: Let $q_a \in Q_d$, and let $c_{q_a,i} \in q_a$ be the ith component of q_a. By construction

$$\Phi_{ij}(q_a, e_o) = Pr(c_{q_b,j}, e_o \mid c_{q_a,i}) \tag{47}$$

$$\Omega_{ij}(q_a, q_b) = \sum_{e_o \in \Sigma_o} Pr(c_{q_b,j}, e_o \mid c_{q_a,i}). \tag{48}$$

The sum of the ith row of $\Omega(q_a, q_b)$ is, therefore, given by

$$\sum_{j=1}^{\|q_b\|} \Omega_{ij}(q_a, q_b) = \sum_{j=1}^{\|q_b\|} \sum_{e_o \in \Sigma_o} Pr(c_{q_b,j}, e_o \mid c_{q_a,i}). \tag{49}$$

When the matrix $\Pi(G_d)$ is constructed, the kth row of $\Pi(G_d)$ is constructed from the ith rows of $\Omega_{q_a,q}$, where q is an arbitrary logical element in the stochastic diagnoser. So, the sum of the kth row of $\Pi(G_d)$ is given by

$$\sum_{q \in Q_d} \sum_{j=1}^{\|q\|} \Omega_{ij}(q_a, q) = \sum_{q \in Q_d} \sum_{j=1}^{\|q\|} \sum_{e_o \in \Sigma_o} Pr(c_{q,j}, e_o \mid c_{q_a,i}) \tag{50}$$

$$= \sum_{e_o \in \Sigma_o} Pr(e_o \mid c_{q_a,i}) \tag{51}$$

$$= 1. \tag{52}$$

Therefore, the sum of each row of $\Pi(G_d)$ is 1. The number of rows in $\Pi(G_d)$ is equal to the sum of the number of rows in $\Omega(q_i, q)$ for any $q \in Q_d$, i.e., $\sum_{i=1}^{n} \|q_i\|$. Similarly, the number of columns in $\Pi(G_d)$ is equal to the sum of the number of columns in $\Omega(q, q_i)$ for any $q \in Q_d$, i.e., $\sum_{i=1}^{n} \|q_i\|$. Because it has an equal number of rows and columns, $\Pi(G_d)$ is a square matrix. Therefore, $\Pi(G_d)$ is a Markov transition matrix. ■

Example 2: To illustrate the results of Theorem 2, consider again the stochastic diagnoser shown in Fig. 5. Denote this diagnoser by G_d. The Markov chain embedded in G_d has a matrix of transition probabilities $\Pi(G_d)$ that is given at the bottom of the page. The components to the left of the matrix indicate which component is associated with each row of the matrix. The horizontal and vertical lines in the matrix delineate the boundaries between the different matrices $\Omega(q_i, q_j)$. By inspecting $\Pi(G_d)$, the matrices associated with each transition of G_d can easily be identified.

C. Relevant Results From Markov Chain Theory

Because the components of the diagnoser can be thought of as states of a finite Markov chain, we will be able to use the theory of finite-state Markov chains to derive conditions for A- and AA-diagnosability. The following subsection is a review of the results in finite-state Markov chain theory that will be essential

$$\begin{array}{r|c|cc|cc|ccc|cc|}
(q_1, 0, N) & 0 & .7 & .05 & .2 & .05 & 0 & 0 & 0 & 0 & 0 \\ \cline{2-11}
(q_2, 0, N) & 0 & .7 & .05 & .2 & .05 & 0 & 0 & 0 & 0 & 0 \\
(q_2, 1, F) & 0 & 0 & .5 & 0 & .5 & 0 & 0 & 0 & 0 & 0 \\ \cline{2-11}
(q_3, 2, N) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
(q_3, 2, F) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \cline{2-11}
(q_4, 0, N) & 0 & 0 & 0 & .2 & .05 & .7 & 0 & .05 & 0 & 0 \\
(q_4, 0, F) & 0 & 0 & 0 & 0 & .25 & 0 & .7 & .05 & 0 & 0 \\
(q_4, 1, F) & 0 & 0 & 0 & 0 & .5 & 0 & 0 & .5 & 0 & 0 \\ \cline{2-11}
(q_5, 0, N) & 0 & 0 & 0 & .2 & .05 & .7 & 0 & .05 & 0 & 0 \\
(q_5, 0, F) & 0 & 0 & 0 & 0 & .25 & 0 & .7 & .05 & 0 & 0
\end{array}.$$

for this paper; readers seeking a more thorough review of the subject should consult [29] or [30].

Suppose that x and y are two states of a Markov chain. The notation ρ_{xy} indicates the probability that if the Markov chain is in state x, it will, at some point in the future, visit state y.

If, for a state x, $\rho_{xx} = 1$, that state is called *recurrent*. Otherwise, if $\rho_{xx} < 1$, then x is a *transient* state. If a state is transient, then at some point in the evolution of the Markov chain the system will leave that state and never return; on the other hand, if a state is recurrent, if the Markov chain visits the state once, the chain will return to that state infinitely often. If the Markov chain is finite-state, there must be at least one recurrent state in the chain.

If $\rho_{xy} > 0$, it is said hat y is *reachable* from x; this is denoted by $x \rightarrow y$. If x is a recurrent state and $x \rightarrow y$, then y is also a recurrent state and $y \rightarrow x$. If there is a set of recurrent states $\{x_1, x_2, \ldots, x_n\}$ such that $x_i \rightarrow x_j$ and $x_j \rightarrow x_i$ for $\forall i, j \in \{1, \ldots, n\}$, then that set is called a *recurrence class*. Paz [2] indicates that determining whether or not a state is recurrent is a decidable problem.

Suppose a chain starts in state x. The probability that after n transitions, the state of the Markov chain will have transitioned from x to y is denoted by $P^n(x, y)$. We can rewrite this probability in discrete-event notation as

$$P^n(x,y) = Pr(t : \delta(x,t) = y \mid t \in L(G,x) \wedge \|t\| = n). \tag{53}$$

As the number of transitions in a Markov chain grows large, the probability of being in a transient state approaches zero. As this idea is central to the development of conditions for A- and AA-diagnosability, it will be expressed formally in the following lemma.

Lemma 1: Let $\mathcal{X}$ be the finite state–space of a Markov chain, and let $\mathcal{T} \subset \mathcal{X}$ be the set of transient states of the chain.

Let $x \in \mathcal{X}$ be an arbitrary state of the Markov chain, and let t be an arbitrary sequence of state transitions beginning at x. Then, $\forall \epsilon > 0$, $\exists n \in \mathbb{N}$ such that

$$Pr(t : \delta(x,t) \in \mathcal{T} \mid t \in L(G,x) \wedge \|t\| = n) < \epsilon. \tag{54}$$

Proof: Let $y \in \mathcal{T}$. Then, the number of times that the evolution of the chain takes it to state y will be denoted by $N_x(y)(t)$ the number of times the state $y \in \mathcal{T}$ is visited along a string $t \in L(G,x)$. Then

$$Pr(N_x(y) = m) = \rho_{xy}\rho_{yy}^{m-1}(1 - \rho_{yy}) \tag{55}$$

$$E(N_x(y)) = \sum_{m=1}^{\infty} m\, Pr(N_x(y) = m) \tag{56}$$

$$= \sum_{m=1}^{\infty} m\, \rho_{xy}\rho_{yy}^{m-1}(1 - \rho_{yy}) \tag{57}$$

$$= \frac{\rho_{xy}}{1 - \rho_{yy}} < \infty \tag{58}$$

as the fact that y is a transient state implies that $1 - \rho_{yy} > 0$.

Furthermore

$$\sum_{n=0}^{\infty} P^n(x,y) = E(N_x(y)) < \infty. \tag{59}$$

In order for the sum of an infinite series to be finite, the terms of that infinite series must approach zero. Therefore, $\forall \epsilon > 0$, $\exists n \in \mathbb{N}$ such that

$$\|t\| = n \Rightarrow P^{\|t\|}(x,y) < \frac{\epsilon}{l}. \tag{60}$$

This relationship can be rewritten in discrete-event notation as

$$Pr(t : t \in L(G,x) \wedge \|t\| = n \wedge \delta(x,t) = y) < \frac{\epsilon}{l} \tag{61}$$

where l is the number of transient states of the Markov chain. Since y is an arbitrary element of $\mathcal{T}$, we can conclude

$$Pr(t : \delta(x,t) \in \mathcal{T} \mid t \in L(G,x) \wedge \|t\| = n)$$

$$= \sum_{y \in \mathcal{T}} Pr(t : \delta(x,t) = y \mid t \in L(G,x) \wedge \|t\| = n)) \tag{62}$$

$$< \sum_{y \in \mathcal{T}} \frac{\epsilon}{l} < \epsilon. \tag{63}$$

■

V. Conditions for Diagnosability

In this section, we present necessary and sufficient conditions for a stochastic automaton to be A-diagnosable, and sufficient conditions for a stochastic automaton to be AA-diagnosable. These conditions are expressed in terms of the stochastic diagnoser introduced in the previous section and take advantage of the Markov properties of the diagnoser shown by Theorem 2. Before determining conditions for A- and AA-diagnosability, we present some additional properties of the stochastic diagnoser.

A. Properties of the Stochastic Diagnoser

The following properties of the stochastic diagnoser can be deduced from the properties of the label propagation function and the Markovian structure of the problem.

Property 1: All components that are reachable from a component with the label F_i also bear the label F_i.

Proof: If $F_i \in l_1$, then $F_i \in LP(x, l_1, s)$ for $\forall x \in \mathcal{X}, s \in L_o(G, x)$. Essentially, once a failure label is appended, it cannot be removed. ■

Property 2: A logical element q of the diagnoser is said to be F_i*-certain* if for all $(x, l) \in q$, either $F_i \in l$ or $F_i \notin l$. If a logical element is F_i-certain, then either every string reaching that element contain some event σ such that $\sigma \in \Sigma_{f_i}$ or there does not exist any string reaching that element that contains any such σ.

This property is shown in [1, Lemma 1-i].

Property 3: All components in the same recurrence class have the same failure label.

Proof: Suppose c_1 and c_2 are components in the same recurrence class. Then, c_2 is reachable from c_1 and *vice versa*. From Property 1, if a label were appended to c_2 by a string reaching from c_1 to c_2, it cannot be removed again by any string reaching from c_2 to c_1. Therefore, c_1 and c_2 must carry the same label. ■

Property 4: All components reachable from a recurrent component bearing the label F_i in an F_i-uncertain logical element are contained in F_i-uncertain elements.

Proof: Let c_r denote a recurrent component bearing the label F_i in an F_i-uncertain logical element.

Consider a logical element $q_f \in Q_d$ where all components have labels that include F_i. By Property 1, all components reachable from any component in this element also bear the label F_i. The diagnoser transition function δ_d shows that any logical element reachable from q_f contains only those components that are reachable from the components of q_f. From Property 1, the only components reachable from the components of q_f carry the label F_i; therefore, all logical elements reachable from q_f must be F_i-certain.

Since c_r is in an F_i-uncertain element, it cannot be reached from q_f. Therefore, no component of q_f can be in the same recurrence class as c_r, which implies that no component of q_f is reachable from c_r.

Furthermore, from Property 1, no component that is reachable from c_r cannot carry a label F_i, so no element that is certain that F_i did not occur can be reachable from c_r. Therefore, the only logical elements that can be reached from c_r are F_i-uncertain. ■

B. Necessary and Sufficient Condition for A-Diagnosability

Using the previous properties of the stochastic diagnoser, we can state conditions for a language L to be A-diagnosable in terms of the structure of the diagnoser.

Theorem 3: A language L generated by a stochastic automaton G is F_i-A-diagnosable if, and only if, every logical element of its diagnoser G_d containing a recurrent component bearing the label F_i is F_i-certain.

Proof: Necessity: Necessity will be shown by contradiction. Suppose there exists $q \in Q_d$ that such that q is not F_i-certain and q contains a recurrent component $c_f = (q, x, l_f)$ such that $F_i \in l_f$. We will then show that

$$(\exists \epsilon > 0)(\exists s \in \Psi(\Sigma_{f_i}))(\exists N \in \mathbb{N})(\forall n \geq N) \left\{ Pr\left(t : D(st) = 0 \mid t \in \frac{L}{s} \wedge \|t\| = n\right) > \epsilon \right\}. \tag{64}$$

By construction, every component of every logical element of the diagnoser is accessible from the initial element. Therefore, there exists a string st, where $s \in \Psi(\Sigma_{f_i})$ and $t \in L/s$ such that $\delta_{\text{comp}}(c_o, st) = c_f$ and $Pr(t) > 0$. Because c_f is in an F_i-uncertain logical element, $D(st) = 0$.

Let $n \in \mathbb{N}$. Let $u \in L/st$ be such that $\|tu\| \geq N$. By Property 4, for $\forall u \in L/st$, the string u transitions the diagnoser to an F_i-uncertain element. Therefore, for all $n \geq N$

$$Pr\left(u : D(stu) = 0 \mid tu \in \frac{L}{s} \wedge \|tu\| = n\right) = 1. \tag{65}$$

Choose $\epsilon > 0$ such that

$$0 < \epsilon < Pr\left(t \mid tu \in \frac{L}{s}\right). \tag{66}$$

Equations (65) and (66), along with the fact that $Pr(tu \mid tu \in L/s) = Pr(t \mid tu \in L/s)Pr(u \mid tu \in L/s)$, imply that for $\forall n \in \mathbb{N}$

$$\begin{aligned} &Pr\left(tu : D(stu) = 0 \mid tu \in \frac{L}{s} \wedge \|tu\| = n\right) \\ &= Pr\left(t \mid tu \in \frac{L}{s} \wedge \|tu\| = n\right) \\ &\quad Pr\left(u : D(stu) = 0 \mid tu \in \frac{L}{s} \wedge \|tu\| = n\right) \\ &> \epsilon. \end{aligned} \tag{67}$$

Therefore, if there is a recurrent component carrying the label F_i in an F_i-uncertain logical element, the stochastic automaton is not A-diagnosable.

Sufficiency: Let $\mathcal{C}$ be the set of components of a stochastic diagnoser, and let $\mathcal{T}_c \in \mathcal{C}$ be the set of transient components. Suppose that every $q \in Q_d$ that contains a recurrent component (q, x, l_f) such that $\Sigma_{f_i} \in l_f$ is F_i-certain.

Let $s \in \Psi(\Sigma_{f_i})$. By Lemma 1, there exists $n \in \mathbb{N}$ such that $\forall c = (q, x, l) \in \mathcal{C}$

$$Pr(t : \delta_{\text{comp}}(c, t) \in \mathcal{T}_c \mid \|t\| = n \wedge t \in L(G, x)) < \epsilon. \tag{68}$$

Since $\delta(x_0, s)$ is a component of the diagnoser of the system reached by s, this implies that

$$Pr\left(t : \delta_{\text{comp}}(c, t) \in \mathcal{T}_c \mid \|t\| = n \wedge t \in \frac{L}{s}\right) < \epsilon. \tag{69}$$

Therefore, if at least n events have occurred since the failure event, with probability greater than $1 - \epsilon$, we will reach an element that contains at least one recurrent component.

However, because $s \in \Psi(\Sigma_{f_i})$, $F_i \in LP(x_o, N, s)$. From Property 1, any label reachable after the string s must contain F_i.

If the true behavior of the system reaches a recurrent component with label F_i, then, by assumption, that component is part of an F_i-certain logical element. Therefore $D(st) = 1$.

Since the probability of reaching an F_i-certain element is at least $1 - \epsilon$

$$Pr\left(t : D(st) = 1 \mid t \in \frac{L}{s} \wedge \|t\| = n\right) > 1 - \epsilon \tag{70}$$

$$Pr\left(t : D(st) = 0 \mid t \in \frac{L}{s} \wedge \|t\| = n\right) < \epsilon. \tag{71}$$

Therefore, if every logical element containing a recurrent component bearing the label F_i is F_i-certain, the system is A-diagnosable. ■

Example 3: Fig. 6 shows the stochastic diagnoser of the stochastic automaton in Fig. 3, which was shown in Section III to be A-diagnosable, but not diagnosable according to Definition 1.

The conditions for logical diagnosability indicate that this stochastic automaton is not logically diagnosable because there

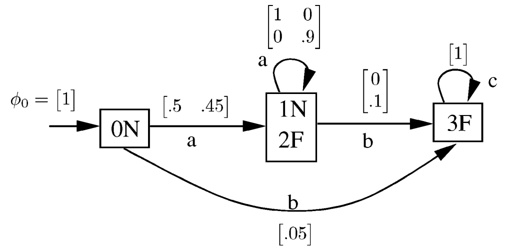
$\begin{bmatrix}1 & 0\\ 0 & .9\end{bmatrix}$
a
[1]
c
$\phi_0 = [1]$
[.5 .45]
0N
a
1N
2F
$\begin{bmatrix}0\\ .1\end{bmatrix}$
b
3F
b
[.05]

Fig. 6. Diagnoser of the system in Fig. 3. The recurrent components of the system are $(q_2, 1, N)$ and $(q_3, 3, F)$.

is a string a^n that takes the diagnoser into a cycle of F-uncertain logical elements, and this cycle of F-uncertain elements corresponds to two separate cycles in the original system model.

However, the Markov matrix associated with this stochastic diagnoser is

$$\begin{array}{c} (q_1, 0, N) \\ (q_2, 1, N) \\ (q_2, 2, F) \\ (q_3, 3, F) \end{array} \begin{bmatrix} 0 & .5 & .45 & .05 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & .9 & .1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

From this matrix, we can determine that the recurrent components are $(q_2, 1, N)$ and $(q_3, 3, F)$. To test for A-diagnosability, we need only consider the recurrent component carrying the label F. This component is the only component in the logical element q_3; therefore it is part of an F-certain element. Therefore, the stochastic automaton in Fig. 3 is A-diagnosable.

Theorem 3 indicates that in order to test for A-diagnosability, we only need to be able to determine the recurrent components of the stochastic diagnoser. This is equivalent to determining the recurrent states of a Markov chain. Therefore, in order to test for A-diagnosability, we need only know which events in our model of the system have a nonzero probability of occurring and we do not need to know the specific values of $p(x', e \mid x)$. This allows us to confirm if our system is A-diagnosable even if we have not modeled the transition probabilities exactly.

In the logical diagnosability conditions considered in [1], the diagnoser itself was insufficient to confirm whether or not a system was diagnosable; in general, it was also necessary to consult a nondeterministic generator automaton based on the logical automaton. However, the stochastic diagnoser contains sufficient information to test the necessary and sufficient conditions for A-diagnosability without reference to a generator or to the original stochastic automaton. The reason for this is that the set of matrices Φ contains information that is lost in the construction of the logical diagnoser; specifically, whether certain components in one diagnoser state are reachable for certain components in other diagnoser states. This additional feature of the stochastic diagnoser captures the information contained in the generator automaton and allows the calculation of recurrent components to be made by consulting only the stochastic diagnoser.

C. A Sufficient Condition for AA-Diagnosability

Again using the properties of the stochastic diagnoser developed in Section V-A, we can determine a condition sufficient to guarantee AA-diagnosability.

Theorem 4: A language L generated by a stochastic automaton G is F_i-AA-diagnosable if for every logical element in the diagnoser G_d constructed from G, the set of recurrent components is F_i-certain.

Proof: Suppose that in each logical element of the diagnoser, the set of recurrent components is F_i-certain. Let $\mathcal{C}$ be the set consisting of every component in every element in the stochastic diagnoser, and let $\mathcal{T}_c \subset \mathcal{C}$ and $\mathcal{R}_c \subseteq \mathcal{C}$ be the sets of transient and recurrent components of the diagnoser, respectively.

From Theorem 2, the components of the stochastic diagnoser can be treated as states of a Markov chain. Therefore, we can apply Lemma 1 to say that $\forall \epsilon > 0$, $\alpha < 1$, $\exists N \in \mathbb{N}$ such that $\forall n > N$

$$Pr\left(st : \delta_{\text{comp}}(q, x, l, t) \in \mathcal{T}_c \mid t \in \frac{L}{s} \wedge \|t\| = n\right) < \epsilon(1-\alpha) \tag{72}$$

for any $s \in L$. For simplicity of notation, we will denote $Pr(. \mid t \in L/s \wedge \|t\| = n)$ by $\hat{P}(.)$.

Suppose we observe the a string $s_o t_o \in \Sigma_o^*$. We can then condition the probability in (72) on $s_o t_o$, yielding

$$\sum_{s_o t_o} \hat{P}(st : \delta_{\text{comp}}(q, x, l, t) \in \mathcal{T}_c \mid s_o t_o) \hat{P}(s_o t_o) < \epsilon(1 - \alpha). \tag{73}$$

Because every term in this summation is nonnegative, we can consider only the subset of possible strings where $\hat{P}(st : \delta_{\text{comp}}(q, x, l, t) \in \mathcal{T}_c \mid s_o t_o) > 1 - \alpha$. For convenience of notation, that conditional probability will be denoted by U, and the conditional probability $\hat{P}(st : \delta_{\text{comp}}(q, x, l, t) \in \mathcal{R}_c \mid s_o t_o) > \alpha$ will be denoted by $\bar{U}$. This results in the following derivation:

$$\sum_{s_o t_o : U} \hat{P}(st : \delta_{\text{comp}}(q, x, l, t) \in \mathcal{T}_c \mid s_o t_o) \hat{P}(s_o t_o) < \epsilon(1 - \alpha) \tag{74}$$

$$\sum_{s_o t_o : U} \frac{\hat{P}(st : \delta_{\text{comp}}(q, x, l, t) \in \mathcal{T}_c \mid s_o t_o)}{1 - \alpha} \hat{P}(s_o t_o) < \epsilon \tag{75}$$

$$\sum_{s_o t_o : U} \hat{P}(s_o t_o) < \epsilon \tag{76}$$

$$\sum_{s_o t_o : U} \hat{P}(s_o t_o) < \epsilon \tag{77}$$

$$\sum_{s_o t_o : \bar{U}} \hat{P}(s_o t_o) \geq 1 - \epsilon. \tag{78}$$

Now, consider a fixed $s \in \Psi(\Sigma_{f_i})$. Since $F_i \in LP(x_0, N, s)$, any recurrent component reachable by st bears the label F_i. Therefore, if the probability that the system is in a recurrent component is greater than α, the probability that F_i has occurred is also greater than α, under the assumption that the set of recurrent components in each logical element is F_i-certain. That is to say, if the failure has occurred, the condition $\hat{P}(st : \delta_{\text{comp}}(q, x, l, t) \in \mathcal{R}_c \mid s_o t_o) > \alpha$ implies that $D_\alpha(st) = 1$ by (25).

Therefore, if $s \in \Psi(\Sigma_{f_i})$, $t \in L/s$ and $\|t\| \geq N$, we can rewrite (78) as

$$\sum_{s_o t_o : D_\alpha(st)=1} \hat{P}(s_o t_o) \geq 1 - \epsilon \tag{79}$$

$$\sum_{s_o t_o : D_\alpha(st)=0} \hat{P}(s_o t_o) < \epsilon. \tag{80}$$

Therefore, for $\forall s \in \Psi(\Sigma_{f_i})$, if the true continuation is n events long ($n \geq N$), the probability of the set of strings of observable events that take the stochastic diagnoser to an information state where diagnosis can not be made has a probability of less than ϵ. Therefore

$$Pr\left(t : D_\alpha(st) = 0 \mid t \in \frac{L}{s} \wedge \|t\| = n\right) < \epsilon. \tag{81}$$

That is to say, if the set of recurrent components in each logical element of the stochastic diagnoser is F_i-certain, the system i F_i-diagnosable. ■

Example 4: 4 The stochastic diagnoser of the system in Fig. 4 is shown in Fig. 7. In Section III-B, it was shown that this stochastic automaton is AA-diagnosable but not A-diagnosable. The Markov matrix associated with its stochastic diagnoser is

$$\begin{array}{c} (q_1,0,N) \\ (q_2,1,N) \\ (q_3,0,N) \\ (q_3,0,F) \\ (q_4,1,N) \\ (q_4,1,F) \end{array} \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & .9 & .1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & .9 & .1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

Inspection of the matrix reveals that the recurrent components of the stochastic diagnoser are $(q_3, 0, F)$ and $(q_4, 1, F)$. Because these components appear in logical elements that are not F-certain, this system is not A-diagnosable.

However, the component $(q_3, 0, F)$ is the only recurrent component in the logical element q_3, and $(q_4, 1, F)$ is the only recurrent component in q_4. Thus, the set of recurrent components in each logical element is F-certain and therefore the system is AA-diagnosable. For this particular system, there is no string of finite length such that the probability of failure given that string is 1, but the probability of failure approaches 1 as the length of the observed string increases.

As is the case with A-diagnosability, it is not necessary to refer to the original stochastic automaton to determine if a system meets this sufficient condition for AA-diagnosability. The set of matrices Φ provides enough information to determine which components are recurrent, thereby making it unnecessary to refer to the original system model.

Although this condition developed in Theorem 4 is sufficient for AA-diagnosability, it is not necessary. The next example will show a system that is AA-diagnosable but does not meet the condition of Theorem 4.

Example 5: Consider the system in Fig. 8. We can use the observations after the occurrence of the failure to determine if the system is in states 1 or 2.

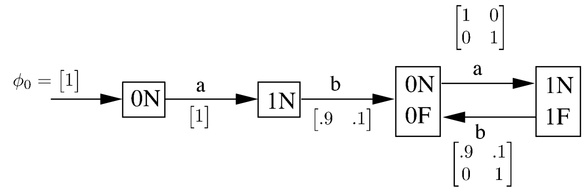
$\phi_0 = [1]$
0N
a
[1]
1N
b
[.9 .1]
0N
0F
a
1N
1F
b
$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
$\begin{bmatrix} .9 & .1 \\ 0 & 1 \end{bmatrix}$

Fig. 7. Stochastic diagnoser of the system shown in Fig. 4. This system is AA-diagnosable but not A-diagnosable.

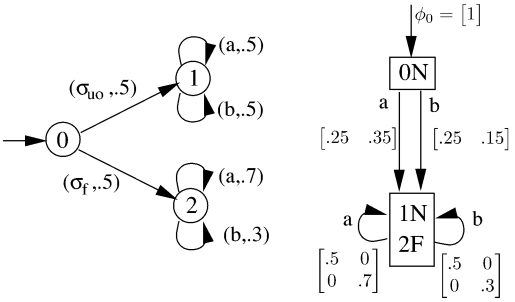

Fig. 8. System that is AA-diagnosable, but does not satisfy the sufficient condition of Theorem 4. States 1 and 2 can be distinguished by observing the relative frequency of the events a and b.

The AA-diagnosability of the system in Fig. 8 can be determined using hypothesis testing techniques [31]. Because states 1 and 2 are separate recurrence classes, we can treat each state as a hypothesis for the true state of the system.

Let H_i denote the hypothesis that the system is in state i, $i = 1, 2$. If we observe n events, a certain fraction will be a and the rest will be b. Let $\hat{a}_n$ denote the fraction of the n observed events that are a.

To determine which hypothesis is correct, we consider the likelihood function $L(H_1 \mid \hat{a}_n)$

$$\begin{aligned} L(H_1 \mid \hat{a}_n) &= \frac{Pr(H_1 \mid \hat{a}_n)}{Pr(H_2 \mid \hat{a}_n)} \\ &= \frac{Pr(\hat{a}_n \mid H_1) Pr(H_1)}{Pr(\hat{a}_n \mid H_2) Pr(H_2)} \\ &= \frac{\binom{n}{\hat{a}_n n} (.7)^{\hat{a}_n n} (.3)^{(n - \hat{a}_n n)}}{\binom{n}{\hat{a}_n n} (.5)^{\hat{a}_n n} (.5)^{(n - \hat{a}_n n)}} \\ &= (1.4)^{\hat{a}_n n} (0.6)^{(n - \hat{a}_n n)}. \end{aligned} \tag{82}$$

Taking the logarithm of the likelihood function gives

$$\begin{aligned} \log L(H_1 \mid \hat{a}_n) &= \hat{a}_n n \log(1.4) + (n - \hat{a}_n n) \log(0.6) \\ &= n(\log(0.6) + \hat{a}_n(\log(1.4) - \log(0.6))). \end{aligned} \tag{83}$$

As n grows large, the log-likelihood of H_1 grows large as well, provided the term $\log(0.6) + \hat{a}_n(\log(1.4) - \log(0.6))$ is greater than zero, which is the case when $\hat{a}_n > .61$.

If the failure has occurred and the state of the system is state 2, we can determine from the law of large numbers that

$$(\forall \epsilon > 0)(\exists n_1 \in \mathbb{N}) \text{ s.t. } n \geq n_1 \Rightarrow Pr(|\hat{a}_n - 0.7| > .09) < \epsilon.$$

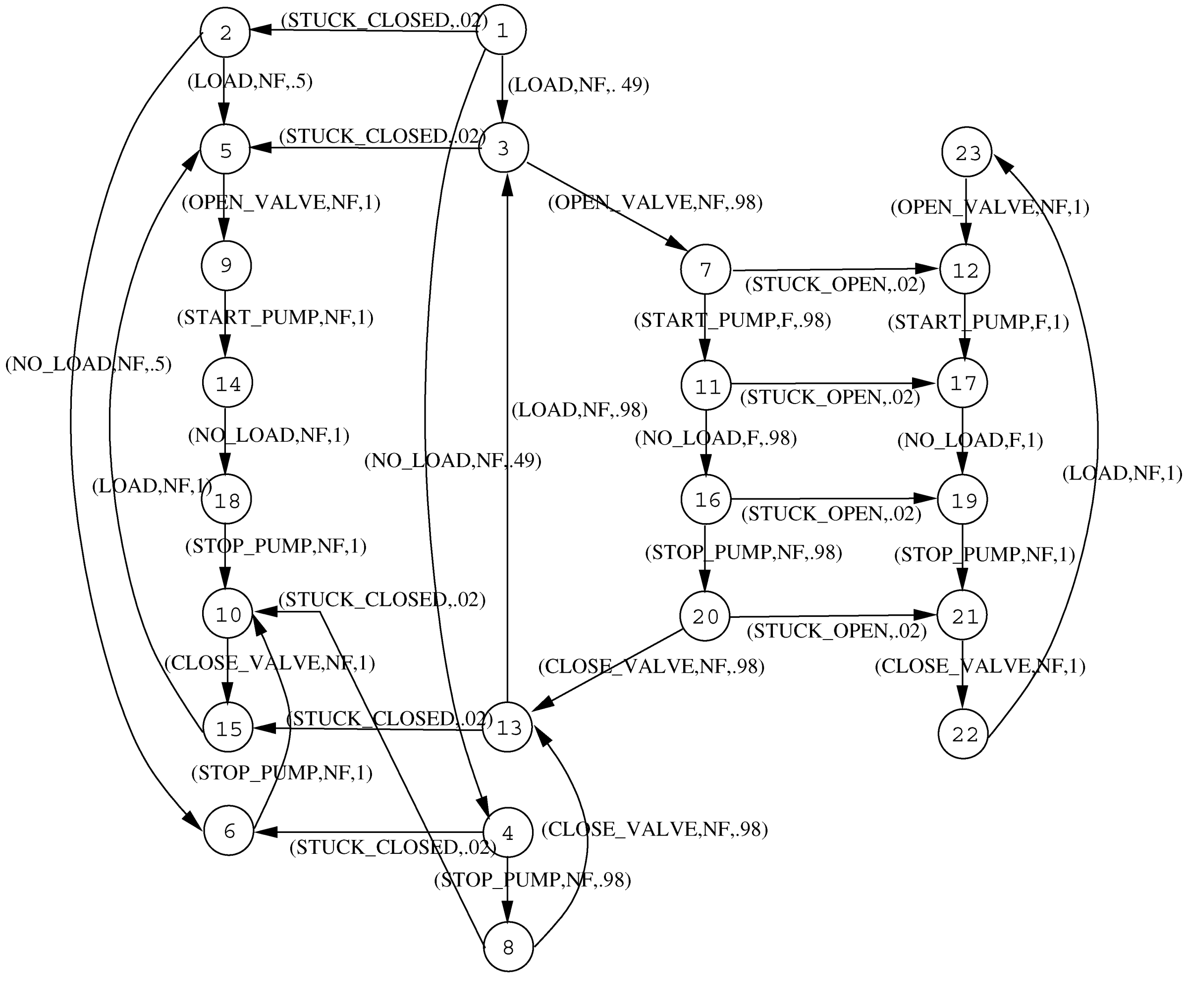

Fig. 9. HVAC system to be diagnosed. The failure events are $\Sigma_{f_1} = \{\mathrm{STUCK\_CLOSED}\}$ and $\Sigma_{f_2} = \{\mathrm{STUCK\_OPEN}\}$.

Therefore, $Pr(\hat{a}_n < .61) < \epsilon$. Let $0 < \alpha < 1$. Now, choose $n_2 \in \mathbb{N}$ such that

$$\begin{aligned}(n \geq n_2) \wedge (\hat{a}_n > .61) &\Rightarrow \log L(H_1 \mid \hat{a}_n) > \log \frac{\alpha}{1-\alpha} \\ &\Rightarrow Pr(H_1 \mid \hat{a}_n) > \alpha.\end{aligned}$$

Let $n = \max(n_1, n_2)$. If a string st occurs such that $\|st\| = n$ and $\hat{a}_n(st) > .61$, then $D_\alpha(st) = 1$. Also, the probability that $\hat{a}_n > .61$ is, by the law of large numbers, greater than $1 - \epsilon$. Therefore, by Definition 3, this system is AA-diagnosable, despite the fact that its stochastic diagnoser contains a state whose recurrent labels are F_i-uncertain.

VI. Examples

We illustrate the results on A- and AA-diagnosability to stochastic versions of models of HVAC systems found in [32]. These examples also illustrate the relationships between logical, A-, and AA-diagnosability.

A. HVAC System

The system model is a stochastic automaton constructed from the composition of four components: a pump, a valve, a controller, and a flow sensor. Probabilities are assigned to each transition in the composed model, producing the stochastic automaton G shown in Fig. 9. This is the system that will be diagnosed.

There are two failures under consideration in this system: the valve in the HVAC system may become stuck open, or it may become stuck closed. Formally, we define $\Sigma_{f_1} = \{\mathrm{STUCK\_CLOSED}\}$ and $\Sigma_{f_2} = \{\mathrm{STUCK\_OPEN}\}$. Using this fault partition, we construct the stochastic diagnoser G_d shown in Fig. 10.

By inspecting the diagnoser and the composed system model, we can see that are no F_1-indeterminate cycles (see [1]) in the diagnoser, but there exists an F_2-indeterminate cycle. Therefore, the failure F_1 (STUCK_CLOSED) is logically diagnosable, but the failure F_2 (STUCK_OPEN) in logically nondiagnosable.

To test for A- and AA-diagnosability, we need to determine which components of G_d are transient and which are recurrent. We do this be constructing the Markov chain associated with this diagnoser. For this example, this Markov chain has 29 states, since there are 29 components in the diagnoser.

From the associated Markov chain, we can determine that there is a recurrence class of components bearing the label F_1 and another recurrence class bearing the label F_2. These components are highlighted in Fig. 10.

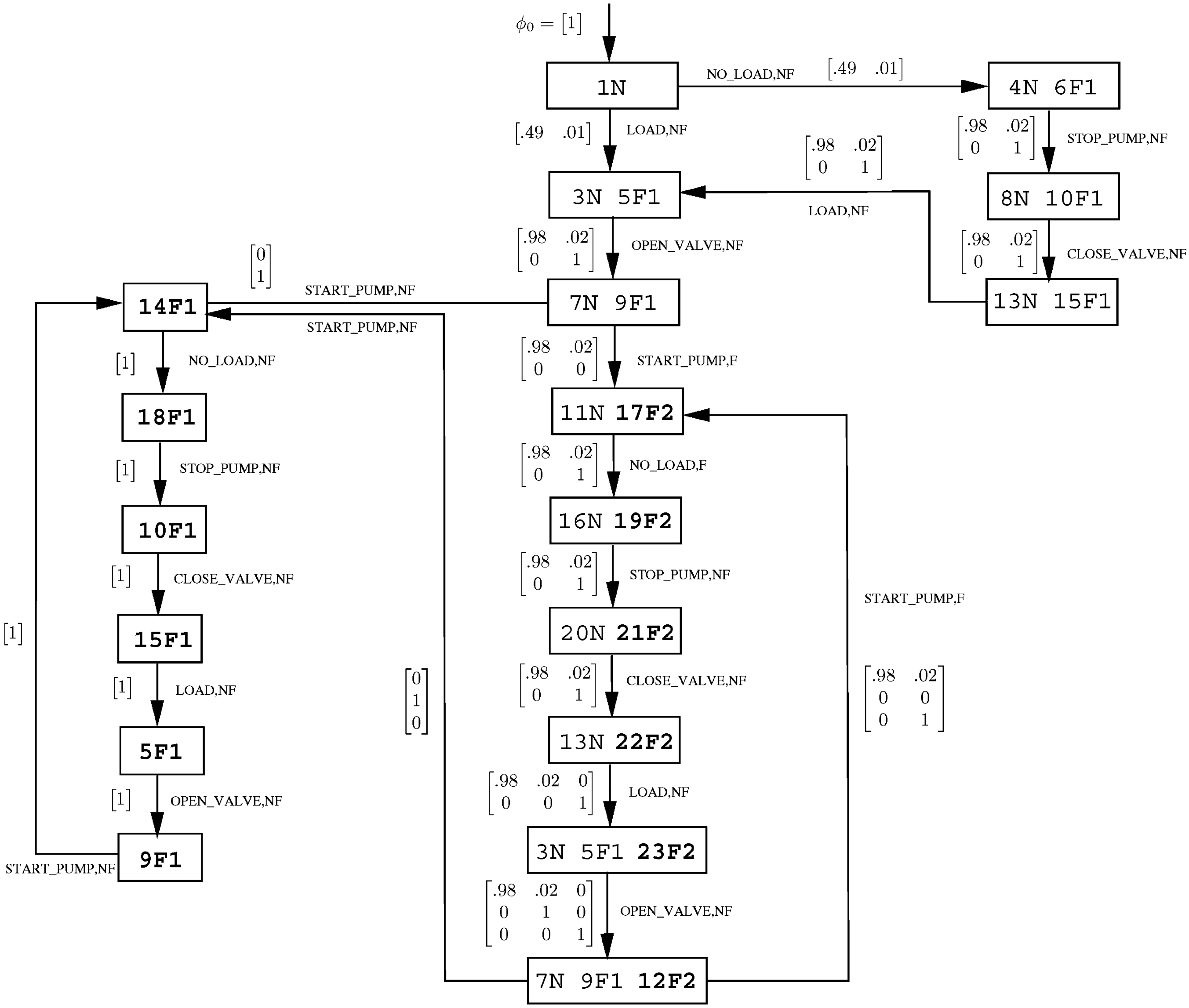

Fig. 10. Stochastic diagnoser of the HVAC system. The recurrent components are highlighted.

Each recurrent component bearing the label F_1 is in a state by itself; therefore, each component is part of an F_1-certain state. We thereby conclude that the fault F_1 is A-diagnosable. Similarly, each of these components is a state whose set of recurrent components if F_1-certain, so F_1 is also AA-diagnosable.

The recurrent components bearing the label F_2 are not in F_2-certain states, therefore F_2 is not A-diagnosable. However, since each of these components is the only recurrent component in its state, F_2 is AA-diagnosable.

In this system, the STUCK_CLOSED failure is logically diagnosable and, thus, the STUCK_CLOSED failure is also A- and AA- diagnosable. However, the STUCK_OPEN failure is neither logically diagnosable nor A-diagnosable; it is merely AA-diagnosable.

This example illustrates that logical diagnosability implies A-diagnosability, which in turn implies AA-diagnosability; however, a system that is AA-diagnosable may be neither logically diagnosable nor A-diagnosable.

Consider the behavior of the stochastic diagnoser along the trace

$$\begin{aligned} s_o = {} & \text{LOAD, OPEN\_VALVE, START\_PUMP, F} \\ & \text{NO\_LOAD, F, STOP\_PUMP, F, CLOSE\_VALVE, NF.} \end{aligned}$$

The logical element reached by s_o is $\{(13, N), (22, F2)\}$. Calculating the probability vector along s_o indicates that the probability that the valve is stuck open s_o is 0.078, as shown by the following calculation:

$$\phi_{un}(s_o) = [1]\,[.49 \quad .01] \begin{bmatrix} .98 & .02 \\ 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} .98 & .02 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} .98 & .02 \\ 0 & 1 \end{bmatrix}^3 = [0.443 \quad 0.037]$$

$$\phi(s_o) = [0.922 \quad 0.078].$$

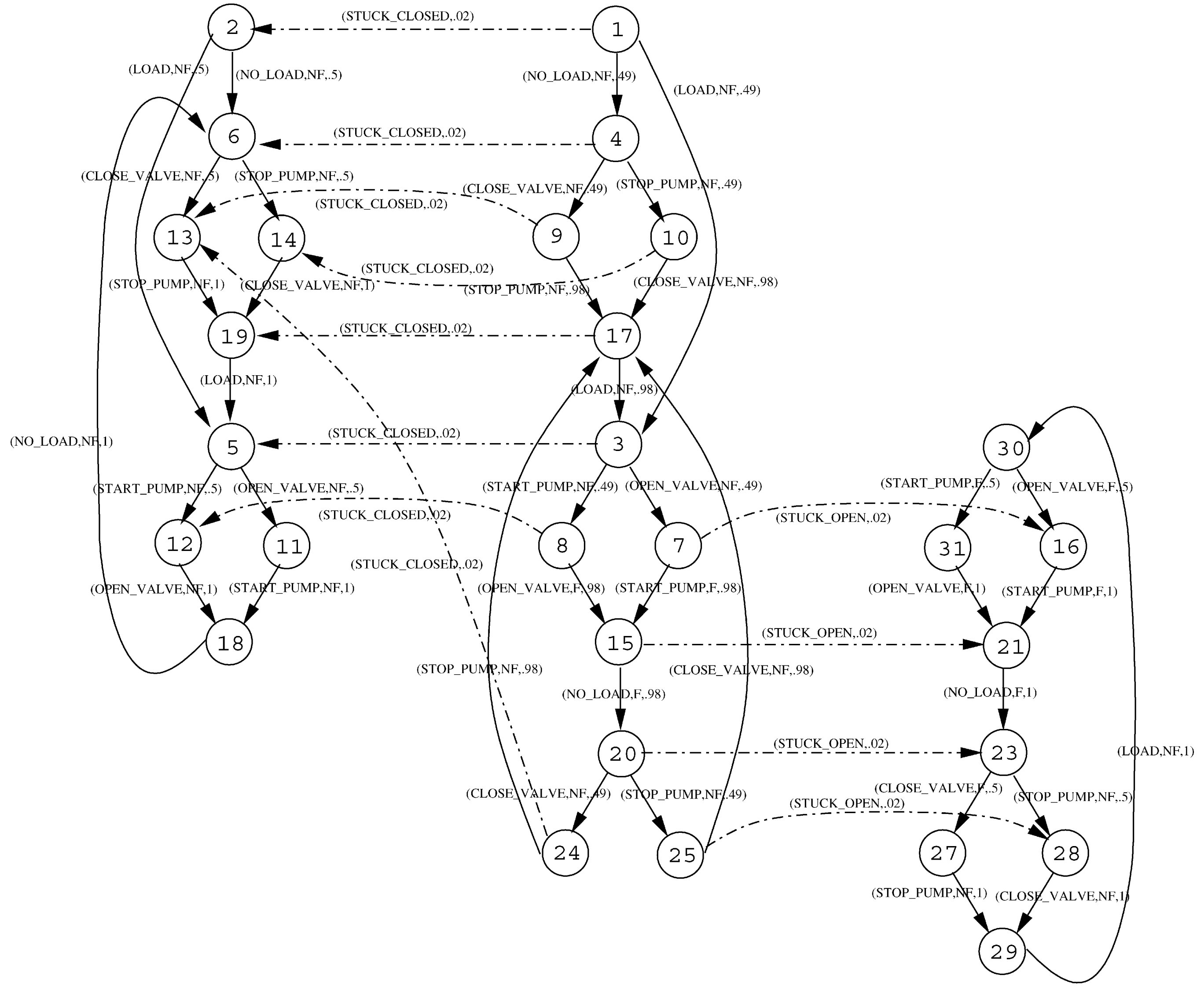

Fig. 11. HVAC system with the new controller. The failure events are $\Sigma_{f_1} = \{\mathrm{STUCK\_CLOSED}\}$ and $\Sigma_{f_2} = \{\mathrm{STUCK\_OPEN}\}$.

B. HVAC System With an Improved Controller

We now modify the stochastic automaton under consideration by replacing the controller from the previous example with another controller, which randomly decides the order in which: a) the STOP_PUMP and CLOSE_VALVE commands are executed; and b) the order in which the START_PUMP and OPEN_VALVE commands are executed. The stochastic automaton composed from this controller, the pump, and the valve is shown in Fig. 11.

As in the previous example, we consider that the valve may became either stuck open or stuck closed and formally define $\Sigma_{f_1} = \{\mathrm{STUCK\_CLOSED}\}$ and $\Sigma_{f_2} = \{\mathrm{STUCK\_OPEN}\}$. The stochastic diagnoser associated with this composed system is shown in Fig. 12.

Once again, inspection of this diagnoser and the original system indicate the absence of any F_1-indeterminate cycles, but the presence of an F_2-indeterminate cycle. Thus, F_1 is again logically diagnosable but F_2 is again logically nondiagnosable.

The Markov chain associated with this stochastic diagnoser contains 43 states; the recurrent components are highlighted in Fig. 12. As before, all recurrent components bearing the label F_1 are in F_1-certain states and, thus, F_1 is both A- and AA-diagnosable.

All the recurrent components bearing the label F_2 are in F_2-certain states. Therefore F_2 is A- and AA-diagnosable.

This example again illustrates that logical diagnosability implies A- and AA-diagnosability. It also shows that a fault may be A-diagnosable without being logically diagnosable, and the an A-diagnosable fault must also be AA-diagnosable.

The previous examples illustrate how the concepts of A- and AA-diagnosability can be applied to practical systems. In the second HVAC example, the controller is an improvement over the first example is the sense that the STUCK_OPEN fault F_2 becomes A-diagnosable when the second controller is used, as opposed to being merely AA-diagnosable under the first controller. Thus, under the second controller, whenever F_2 is diagnosed we can be certain that valve is indeed stuck open; in the first example, we cannot diagnose F_2 online without first setting a probability threshold strictly less than one and thus creating the possibility of a false positive diagnosis. However, sense this

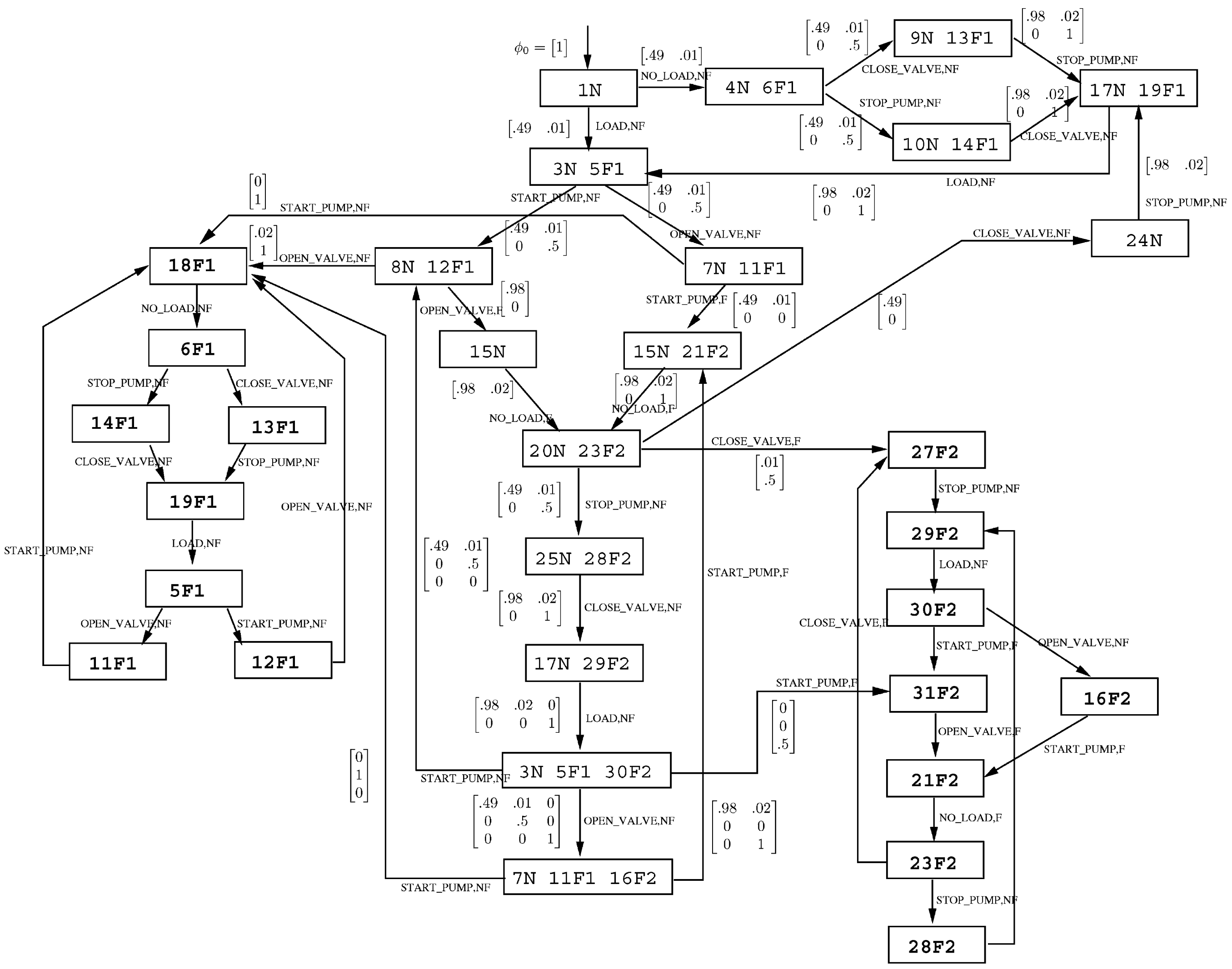

Fig. 12. Stochastic diagnoser of the HVAC system in Fig. 11. The recurrent components are highlighted.

probability threshold can be set arbitrarily close to one, the probability of a false positive can be made arbitrarily small.

VII. Summary and Conclusion

We presented an approach to diagnosability of stochastic automata. We defined A- and AA-diagnosability, two notions of diagnosability that are appropriate for stochastic automata and showed that they are weaker than the corresponding notion of diagnosability for logical automata introduced in [1]. Roughly speaking, a system is A-diagnosable if, in the long run, it is almost sure that we will become certain as to whether or not a failure has occurred. The notion of AA-diagnosability is weaker than A-diagnosability as, in AA-diagnosable systems, it is not necessary that failures be diagnosed with absolute certainty, but merely with almost sure certainty.

We determined offline conditions that ensure diagnosability and showed how a "stochastic diagnoser" can be used for online diagnosis of failure events. An important open problem is the determination of a condition necessary and sufficient to ensure AA-diagnosability.

References

[1] 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.

[2] A. Paz, *Introduction to Probabilistic Automata*. New York: Academic, 1971.

[3] S. Lafortune, D. Teneketzis, M. Sampath, R. Sengupta, and K. Sinnamohideen, "Failure diagnosis of dynamic systems: An approach based on discrete event systems," in *Proc. 2001 Amer. Control Conf.*, Jun. 2001, pp. 2058–2071.

[4] S. Jiang, R. Kumar, and H. Garcia, "Diagnosis of repeated failures in discrete event systems," in *Proc. 41st IEEE Conf. Decision and Control*, Dec. 2002, pp. 4000–4005.

[5] M. Sampath, S. Lafortune, and D. Teneketzis, "Active diagnosis of discrete-event systems," *IEEE Trans. Autom. Control*, vol. 43, no. 7, pp. 908–929, Jul. 1998.

[6] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Failure diagnosis using discrete-event models," *IEEE Trans. Control Syst. Technol.*, vol. 4, no. 2, pp. 105–124, Mar. 1996.

[7] D. Pandalai and L. Holloway, "Template languages for fault monitoring of discrete event processes," *IEEE Trans. Autom. Control*, vol. 45, no. 5, pp. 868–882, May 2000.

[8] F. Lin, "Diagnosability of discrete event systems and its application," *Discrete Event Dyna. Syst.: Theory Appl.*, vol. 4, no. 2, pp. 197–212, May 1994.

[9] S. Jiang and R. Kumar, "Failure diagnosis of discrete event systems with linear-time temporal logic fault specificatioans," in *Proc. 2002 Amer. Control Conf.*, May 2002, pp. 128–133.

[10] S. Bavishi and E. Chong, "Automated fault diagnosis using a discrete event systems framework," in *Proc. 9th IEEE Int. Symp. Intelligent Control*, 1994, pp. 213–218.

[11] R. Debouk, "Failure diagnosis of decentralized discrete event systems," Ph.D. dissertation, Elec. Eng. Comp. Sci. Dept., University of Michigan, Ann Arbor, MI, 2000.

[12] G. Lamperti and M. Zanella, "Diagnosis of discrete event systems integrating synchronous and asynchronous behvior," in *Proc. 9th Int. Workshop on Principles of Diagnosis (DX'99)*, 1999, pp. 129–139.

[13] G. Provan and Y.-L. Chen, "Diagnosis of timed discrete event systems using temporal causal networks: Modeling and analysis," in *Proc. 1998 Int. Workshop on Discrete Event Systems (WODES '98)*, Aug. 1998, pp. 152–154.

[14] S. H. Zad, R. Kwong, and W. Wonham, "Fault diagnosis in discrete-event systems: Framework and model reduction," in *Proc. 37th IEEE Conf. Decision and Control*, Dec. 1998, pp. 3769–3774.

[15] E. Garcia, F. Morant, R. Blasco-Giminez, A. Correcher, and E. Quiles, "Centralized modular diagnosis and the phenomenon of coupling," in *Proc. 2002 IEEE Int. Workshop on Discrete Event Systems (WODES '02)*, Oct. 2002, pp. 161–168.

[16] A. Darwiche and G. Provan, "Exploiting system structure in model-based diagnosis of discrete event systems," in *Proc. 7th Annu. Int. Workshop on the Principles of Diagnosis (DX'96)*, Oct. 1996, pp. 95–105.

[17] G. Provan and Y.-L. Chen, "Model-based diagnosis and control reconfiguration for discrete event systems: An integrated approach," in *Proc. 38th IEEE Conf. Decision and Control*, Dec. 1999, pp. 1762–1768.

[18] R. Sengupta, "Discrete-event diagnostics of automated vehicles and highways," in *Proc. 2001 Amer. Control Conf.*, Jun. 2001.

[19] K. Sinnamohideen, "Discrete-event diagnostics of heating, ventilation, and air-conditioning systems," in *Proc. 2001 Amer. Control Conf.*, Jun. 2001, pp. 2072–2076.

[20] N. Viswanadham and T. Johnson, "Fault detection and diagnosis of automated manufacturing systems," in *Proc. 27th IEEE Conf. Decision and Control*, 1988, pp. 2301–2306.

[21] F. Lin, J. Markee, and B. Rado, "Design and test of mixed signal circuits: A discrete event approach," in *Proc. 32th IEEE Conf. Decision and Control*, Dec. 1993, pp. 246–251.

[22] L. Holloway and S. Chand, "Time templates for discrete event fault monitoring in manufacturing systems," in *Proc. 1994 Amer. Control Conf.*, 1994, pp. 701–706.

[23] G. Westerman, R. Kumar, C. Stround, and J. Heath, "Discrete event system approach for delay fault analysis in digital circuits," in *Proc. 1998 Amer. Control Conf.*, 1998, pp. 239–243.

[24] Y. Pencolé, "Decentralized diagnoser approach: Application to telecommunication networks," in *Proc. 11th Int. Workshop on Principles of Diagnosis (DX'00)*, Jun. 2000, pp. 185–192.

[25] J. Lunze and J. Schröder, "State observation and diagnosis of discrete-event systems described by stochastic automata," *Discrete Event Dyna. Syst.: Theory Appl.*, vol. 11, no. 4, pp. 319–369, 2001.

[26] D. Thorsley and D. Teneketzis, "Diagnosability of stochastic automata," in *Proc. 42nd IEEE Conf. Decision and Control*, Dec. 2003, pp. 6289–6294.

[27] C. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Boston, MA: Kluwer, 1999.

[28] P. Kumar and P. Varaiya, *Stochastic Systems: Estimation, Identification, and Adaptive Control*. Upper Saddle River, NJ: Prentice-Hall, 1986.

[29] P. Brémaud, *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues*. New York: Springer-Verlag, 1999.

[30] P. Hoel, S. Port, and C. Stone, *Introduction to Stochastic Processes*. Prospect Heights, IL: Waveland Press, 1987.

[31] D. Snyder, *Random Point Processes*. New York: Wiley, 1975.

[32] M. Sampath, "A discrete event systems approach to failure diagnosis," Ph.D. dissertation, Univ. Michigan, Ann Arbor, MI, 1995.

David Thorsley received the B.E.Sc. degree in electrical engineering from the University of Western Ontario, London, ON, Canada, in 2000, and the M.S. degree in electrical engineering from the University of Michigan, Ann Arbor, in 2002. He is currently working toward the Ph.D. degree in electrical engineering at the University of Michigan.

His research interests include discrete-event systems, stochastic control, and communication networks.

Demosthenis Teneketzis received the diploma in electrical engineering from the University of Patras, Patras, Greece, and the M.S., E.E., and Ph.D. degrees, all in electrical engineering, from the Massachusetts Institute of Technology, Cambridge, in 1974, 1976, 1977, and 1979, respectively.

He is currently a Professor of Electrical Engineering and Computer Science at the University of Michigan, Ann Arbor. In winter and spring 1992, he was a Visiting Professor at the Swiss Federal Institute of Technology, (ETH), Zurich, Switzerland. Prior to joining the University of Michigan, he worked for Systems Control, Inc., Palo Alto, CA, and Alphatech, Inc., Burlington, MA. His research interests are in stochastic control, decentralized systems, queueing and communication networks, stochastic scheduling and resource allocation problems, mathematical economics, and discrete-event systems.