Proceedings of the
47th IEEE Conference on Decision and Control
Cancun, Mexico, Dec. 9-11, 2008

Inference-Based Decentralized Prognosis in Discrete Event Systems

Shigemasa Takai, Member IEEE
Dept. of Information Sc.
Kyoto Inst. of Tech., Kyoto, Japan
takai@kit.ac.jp

Ratnesh Kumar, Fellow IEEE
Dept. of Elec. & Comp. Eng.
Iowa State Univ., Ames, IA
rkumar@iastate.edu

Abstract— **For discrete event systems, we study the problem of predicting failures prior to their occurrence, also referred to as prognosis, in the inference-based decentralized framework where multiple decision-makers interact to come up with the global prognostic decisions. Due to the limited sensing capabilities, each decision-maker is subjected to ambiguities during the process of decision-making. In our prior work [7] we made an observation that such ambiguities are of differing gradations and presented a framework for inferencing over the local control decisions of varying ambiguity levels to arrive at a global control decision. Here we present an inference-based decentralized decision-making framework for prognosis of failures: For each event-trace executed by a system being monitored, each local prognoser issues its own prognostic decision (failure is or is not inevitable, or unsure) tagged with a certain ambiguity level (zero being the minimum) that is computed by assessing the ambiguities of the self and the others. A global prognostic decision is taken to be the "winning" local prognostic decision, i.e., one with the minimum ambiguity level. We characterize the class of systems for which there are no missed detections (all failures can be prognosed prior to their occurrence) and no false alarms (all prognostic decisions are correct) by introducing the notion of N-inference-prognosability, where the parameter N represents the maximum ambiguity level of any winning prognostic decision. An algorithm for verifying N-inference-prognosability is presented. We also show that the notion of coprognosability introduced in [8] is the same as 0-inference-prognosability, and as the parameter N is increased, a larger class of prognosable systems is obtained.**

I. Introduction

Discrete event systems (DESs) are systems with discrete states that evolve in response to certain discrete changes (called events). Many physical systems such as manufacturing systems, communication protocols, reactive software, digital hardware can be modeled as DESs at a certain level of abstraction. The behaviors of a DES consist of all sequences of events (called traces) it can execute starting from its initial state. A system specification is used to identify the set of traces that are desirable. Execution of a trace that violates a specification constitutes a failure. The task of failure prognosis is to predict the occurrence of an impending failure *prior* to its occurrence. This helps provide a time for reacting to an impending failure so that appropriate corrective actions may be initiated prior to its occurrence. Note the contrast with the task of diagnosis which requires the detection of a failure *after* its occurrence.

The problem of prediction of failures prior to their occurrence is an active area of research (see for example [12] and references there in). In the context of DESs, the prediction of a failure based on a statistical analysis was considered in [1]. Due to the statistical nature of the analysis, the issuance of a prediction alert only means a high confidence in a future occurrence of a failure (but not full confidence). To capture the inevitability of a future failure, the notion of indicator traces (which indicate that a failure is guaranteed to occur) was introduced in [4], where their bounded delay detection was also studied. The notion of indicator traces was later used in [5] to formalize the diagnosis of repeated failures in the setting of temporal logic.

[3] formulated the notion of predictability (or prognosability) of failures: Each failure trace must possess a nonfailure prefix such that any indistinguishable trace has the property that a failure is inevitable within a uniformly bounded number of steps. Note while the existence of a uniform bound for the delay of failure detection is essential for defining diagnosability (otherwise a diagnoser may end up waiting for an arbitrarily long period before diagnosing a failure), the existence of a uniform bound within which a failure is guaranteed to occur is not essential for defining prognosability. This observation led us to weaken the definition of prognosability in [8]: Each failure trace must possess a nonfailure prefix such that each indistinguishable trace is an indicator trace. (Recall an indicator trace is one for which a failure is inevitable.) Our work in [8] also provided a test for prognosability that is polynomial in the size of a plant and a nonfailure specification, improving the test given in [3] which is based on an observer construction and is of exponential complexity. In [8], we also introduced the notion of *reaction bound for prognosis* as the earliest time beyond a prognostic decision when a failure can occur, and presented a polynomial complexity algorithm for computing it. A polynomial complexity algorithm for an online prognosis of failures was also presented in [8].

In the setting of [8], a local prognoser issues a prognostic decision only when it is unambiguous about it. Thus, the framework does not involve any inferencing among the local prognosers over their ambiguities, which can be of different levels. It is known through earlier works reported in [9], [10], [13] that adding inferencing can aid the process of decentralized decision-making. While these prior inferencing-based approaches relied on a "single-level" of inferencing, a framework allowing multi-level inferencing was later presented in [7] (for control), [6] (for diagnosis of failures), and [11] (for diagnosis of nonfailures).

In this paper we introduce the inference-based decentralized decision-making framework for the prognosis of failures. The framework supports multiple levels of inferencing over the ambiguities of the self and the others. Each local

This work was supported in part by the National Science Foundation under the grants NSF-ECCS-0424048, NSF-ECCS-0601570, NSF-ECCS-0801763, and NSF-CCF-0811541, and in part by Grant-in-Aid for Scientific Research (C) 18560433.

978-1-4244-3124-3/08/$25.00 ©2008 IEEE

prognoser uses its observations of the system behavior to come up with its prognostic decision together with a grade or level of ambiguity for that decision. A minimum (level-zero) ambiguity decision is issued by a local prognoser when following all traces, producing the same observation as the one received, either a failure is inevitable (so a positive decision is issued) or it is not inevitable (so a negative decision is issued). In general a local prognoser will issue a positive (resp., negative) decision with an ambiguity level N following a certain observation if for each ambiguous trace, there exists another local prognoser that can issue a negative (resp., positive) decision with an ambiguity level at most $N-1$. Note in certain situations it is possible that a local prognostic decision is neither "positive" nor "negative", but "unsure". The global prognostic decision is taken to be the same as a local prognostic decision whose ambiguity level is the minimum. (Such a local decision can be considered to be a "winning" local decision.)

We characterize the class of systems for which there are no missed detections (all failures can be prognosed prior to their occurrence) and no false alarms (all prognostic decisions are correct) by introducing the notion of N-inference-prognosability, where the parameter N represents the maximum ambiguity level of any winning prognostic decision. An algorithm for verifying N-inference-prognosability is presented. We also show that as the parameter N is increased, a larger class of prognosable systems is obtained. Further the notion of coprognosability introduced in [8] is the same as 0-inference-prognosability, implying that even the class of 1-inference-prognosable systems subsumes the class of coprognosable systems introduced in [8].

II. Notation and Preliminaries

We consider a DES modeled by a nondeterministic automaton $G=(X,\Sigma,\alpha,X_0,X_m)$, where X is the set of states, Σ is the finite set of events, a function $\alpha : X\times(\Sigma\cup\{\varepsilon\})\to 2^X$ is the transition function, $X_0\subseteq X$ is the set of initial states, and $X_m\subseteq X$ is the set of marked or accepting states. G is said to be deterministic if the transition function can be written as a partial function $\alpha : X\times\Sigma\to X$ and $|X_0|=1$. Let Σ^* be the set of all finite sequences of events including the empty sequence ε. Elements of Σ^* are called traces, and subsets of Σ^* are called languages. For each trace $s\in\Sigma^*$, $|s|$ denotes its length. For any $m\in\mathcal{N}$, where $\mathcal{N}$ denotes the set of all nonnegative integers, $\Sigma^{\geq m}:=\{s\in\Sigma^* \mid |s|\geq m\}$ denotes the set of all traces with m or more events. The transition function α can be generalized to $\alpha : 2^X\times\Sigma^*\to 2^X$ in a natural way. The generated and marked (or accepted) languages of G are respectively defined as, $L(G):=\{s\in\Sigma^*|\ \alpha(X_0,s)\neq\emptyset\}$, and $L_m(G):=\{s\in\Sigma^*|\ \alpha(X_0,s)\cap X_m\neq\emptyset\}$.

For a trace $s\in\Sigma^*$, the set of all prefixes of s is denoted by $pr(s)$. The notation $t\leq s$ denotes that t is a prefix of s. For a language K, the set of all prefixes of traces in K is defined as $pr(K)=\bigcup_{s\in K}pr(s)$. K is said to be (prefix-)closed if $K=pr(K)$. A language K is said to be *deadlock-free* if for any $s\in pr(K)$, there exists a trace $t\neq\varepsilon$ such that $st\in pr(K)$; otherwise $s\in K$ is called a deadlocking trace of K. The language K after $s\in\Sigma^*$, denoted by $K\backslash s$, is defined as $K\backslash s:=\{t\in\Sigma^*\mid st\in K\}$.

Let $I=\{1,2,\ldots,n\}$ denote the index set of local prognosers that perform the task of prognosis without sharing their observations. We assume that the limited sensing capabilities of the ith local prognoser P_i ($i\in I$) can be represented as the local observation mask, $M_i : \Sigma\cup\{\varepsilon\}\to\Delta_i\cup\{\varepsilon\}$, where Δ_i is the set of locally observed symbols, and $M_i(\varepsilon)=\varepsilon$. The map M_i is generalized to $M_i : \Sigma^*\to\Delta_i^*$ and $M_i : 2^{\Sigma^*}\to 2^{\Delta_i^*}$ in a natural way.

$$\begin{aligned}&(\forall s\in\Sigma^*,\sigma\in\Sigma,H\subseteq\Sigma^*)\\ &\qquad M_i(\epsilon):=\epsilon;\ M_i(s\sigma)=M_i(s)M_i(\sigma);\\ &\qquad M_i(H)=\{M_i(s)\mid s\in H\}.\end{aligned}$$

III. Inference-Based Decentralized Prognosis Framework

Let $L\neq\emptyset$ be a closed language representing the generated language of a plant, and $K\subseteq L$ be a nonempty closed language representing a nonfailure specification language. Traces in $L-K$ are considered failure traces and the task of prognosis is to predict the execution of any trace in $L-K$. Without loss of generality, the plant language L can be taken to be deadlock-free. Otherwise we can extend each deadlocking trace by an unbounded sequence of a newly added event that is unobservable to all prognosers. This will make the language deadlock-free without altering any of the prognosability properties since the newly added event does not produce any observation to any of the prognosers.

We introduce the notion of an inference-based decentralized prognoser that consists of a set of local prognosers and a central decision fusion unit. Let the set $C=\{0,1,\phi\}$ be the set of prognostic decisions, where "1" means a failure is guaranteed to occur in future, "0" means a failure is not guaranteed to occur in future, and "ϕ" represents an "unsure" decision. Each inference-based local prognoser P_i is defined as a map $P_i : M_i(K)\to C\times\mathcal{N}$, where for each $s\in K$,

$$P_i(M_i(s))=(c_i(M_i(s)),n_i(M_i(s))).$$

Here $c_i(M_i(s))\in C$ denotes the prognostic decision of P_i following an observation $M_i(s)\in M_i(K)$, and $n_i(M_i(s))\in\mathcal{N}$ denotes the ambiguity level of the prognostic decision of P_i. Let $n(s)$ be the minimum ambiguity level of local decisions, i.e.,

$$n(s):=\min_{i\in I}n_i(M_i(s)).$$

The decentralized prognoser $\{P_i\}_{i\in I}$ that consists of local prognosers P_i ($i\in I$) issues global prognostic decisions. Formally, $\{P_i\}_{i\in I}$ is defined as a map $\{P_i\}_{i\in I} : K\to C$. For each $s\in K$, the prognostic decision $\{P_i\}_{i\in I}(s)$ is given as follows:

$$\{P_i\}_{i\in I}(s)=\begin{cases}0, & \text{if } \forall i\in I \text{ s.t. } n_i(M_i(s))=n(s):\\ & c_i(M_i(s))=0\\ 1, & \text{if } \forall i\in I \text{ s.t. } n_i(M_i(s))=n(s):\\ & c_i(M_i(s))=1\\ \phi, & \text{otherwise.}\end{cases}$$

In other words, the global prognostic decision is taken to be the same as a local prognostic decision possessing the minimum level of ambiguity.

A useful notion of a decentralized prognoser is the largest ambiguity level $N \in \mathcal{N}$ of any sure decision, and the preservation of surety of a decision with a decrease in the ambiguity level (if a certain ambiguity level decision is "sure", then all lower ambiguity level decisions are also "sure"). We refer to such a prognoser to be "N-inferring".

Definition 1: A decentralized prognoser $\{P_i\}_{i\in I} : K \to C$ is said to be N*-inferring* if the following two conditions hold:

1) $(\forall s \in K)\ \{P_i\}_{i\in I}(s) \neq \phi \Rightarrow n(s) \leq N$,
2) $(\forall s, s' \in K)\ [\{P_i\}_{i\in I}(s) \neq \phi \wedge n(s') \leq n(s)] \Rightarrow \{P_i\}_{i\in I}(s') \neq \phi$.

IV. Existence/Synthesis of Inference-Based Decentralized Prognosers

We introduce the following notions of *boundary* traces (for which a failure in a next step is guaranteed), *indicator* traces (for which a failure in future is guaranteed), and *nonindicator* traces (that are not indicator traces).

Definition 2: Given a pair (L, K) of closed languages with $K \subseteq L$, we define the set of

- *boundary* traces of K with respect to L as, $\partial_L(K) := \{s \in K \mid \{s\}\Sigma \cap (L - K) \neq \emptyset\}$;
- *indicator* traces of K with respect to L as, $\Im_L(K) := \{s \in K \mid \exists m \in \mathcal{N} : L\backslash s \cap \Sigma^{\geq m} \subseteq [L - K]\backslash s\}$;
- *nonindicator* traces of K with respect to L as, $\Upsilon_L(K) := K - \Im_L(K)$.

Note $\Upsilon_L(K) = \{s \in K \mid \forall m \in \mathcal{N}, \exists t \in L\backslash s \cap \Sigma^{\geq m} : st \in K\}$.

Let $N \in \mathcal{N}$ be a given nonnegative integer. In this section we introduce the notion of N*-inference-prognosability* as a necessary and sufficient condition for the existence of an N-inferring prognoser with the following properties.

- There are no *missed detections*, i.e., each failure is prognosed prior to its occurrence:
$$(\forall s \in L - K)(\exists t \in pr(s) \cap K)(\forall u \in [pr(s) \cap K]\backslash t)$$
$$\{P_i\}_{i\in I}(tu) = 1. \quad (1)$$
- There are no *false alarms*, i.e., an incorrect prognostic decision is never issued:
$$(\forall s \in \Im_L(K))\ \{P_i\}_{i\in I}(s) \neq 0; \quad (2)$$
$$(\forall s \in \Upsilon_L(K))\ \{P_i\}_{i\in I}(s) \neq 1. \quad (3)$$

(1) requires the existence of a nonfailure prefix of each failure trace where a prognostic decision "1" is issued which is continued to be held in future (i.e., the prognoser does not change its mind), whereas (2) (resp., (3)) requires that for each indicator (resp., nonindicator) trace a prognostic decision "0" (resp., "1") is not issued.

In order to introduce the notion of N-inference-prognosability, we inductively define a monotonically decreasing sequence $\{(\Im_k, \Upsilon_k)\}_{0\leq k\leq N+1}$ of language pairs as follows:

- Base step:
$$\Im_0 := \Im_L(K),\ \Upsilon_0 := \Upsilon_L(K).$$
- Induction step:
$$\begin{aligned}\Im_{k+1} &:= \Im_k \cap \left(\bigcap_{i\in I} M_i^{-1}M_i(\Upsilon_k)\right),\\ \Upsilon_{k+1} &:= \Upsilon_k \cap \left(\bigcap_{i\in I} M_i^{-1}M_i(\Im_k)\right).\end{aligned}$$

The computation of the sequence $\{(\Im_k, \Upsilon_k)\}_{0\leq k\leq N+1}$ of language pairs starts with $\Im_0 = \Im_L(K)$, the set of indicator traces, and $\Upsilon_0 = \Upsilon_L(K)$, the set of nonindicator traces. Note that $\Im_{k+1}$ is a sublanguage of $\Im_k$ consisting of those traces for which for each $i \in I$ there exists an M_i-indistinguishable trace in Υ_k. As a result when the plant executes a trace in $\Im_{k+1}$ all the local prognosers will be ambiguous as to whether the executed trace is in $\Im_{k+1}$ or in Υ_k. The sublanguage Υ_{k+1} of Υ_k can be understood in a similar fashion. The language $\Im_{k+1}$ has the following intuitive interpretation: It consists of those traces for which the prognostic decision "1" is required but all prognosers remain ambiguous about it even after k-levels of inferencing. A dual interpretation exists for the language Υ_{k+1}.

Using the sequence $\{(\Im_k, \Upsilon_k)\}_{0\leq k\leq N}$ of language pairs, a local prognoser computes its prognostic decision and associates a level of ambiguity with such a decision as follows. For each $s \in K$, the ith local prognoser P_i computes

$$\begin{aligned}n_i^{\Im}(M_i(s)) &:= \min\{k \mid [M_i(s) \notin M_i(\Upsilon_k)]\\ &\qquad \vee [k = N + 1]\}, \quad (4)\\ n_i^{\Upsilon}(M_i(s)) &:= \min\{k \mid [M_i(s) \notin M_i(\Im_k)]\\ &\qquad \vee [k = N + 1]\}. \quad (5)\end{aligned}$$

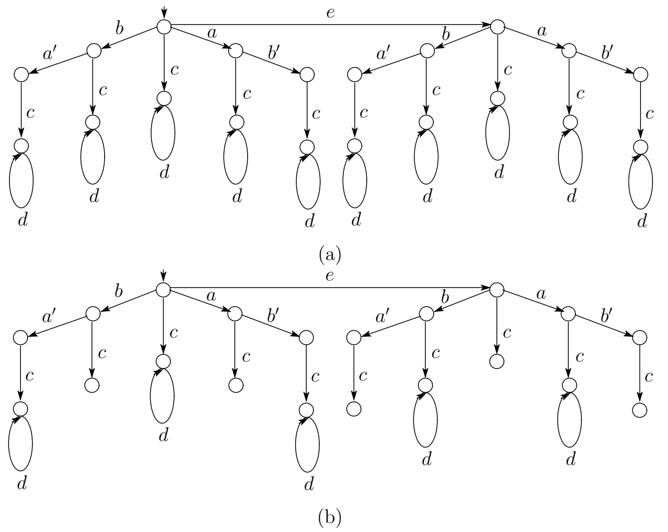
Note that $n_i^{\Im}(M_i(s))$ and $n_i^{\Upsilon}(M_i(s))$ are bounded above by $N + 1$. Here $n_i^{\Im}(M_i(s))$ represents the ambiguity level of a failure prognostic decision "contemplated" by the ith prognoser following the observation $M_i(s)$. (When $n_i^{\Im}(M_i(s)) < N + 1$, it denotes the minimum index k such that the observation $M_i(s)$ does not match with the observations of any of the traces in Υ_k.) Similarly, the notation $n_i^{\Upsilon}(M_i(s))$ represents the ambiguity level of a nonfailure prognostic decision "contemplated" by the ith prognoser following the observation $M_i(s)$. Which of the two contemplated decisions is ultimately issued is decided by comparing the two ambiguity levels, $n_i^{\Im}(M_i(s))$ vs. $n_i^{\Upsilon}(M_i(s))$, and favoring the smaller one. This is formalized next.

For a local prognoser $P_i : M_i(K) \to C \times \mathcal{N}$, its prognostic decision and ambiguity level following an observation $M_i(s) \in M_i(K)$, i.e., $P_i(M_i(s)) = (c_i(M_i(s)), n_i(M_i(s)))$, is determined as follows:

$$c_i(M_i(s)) = \begin{cases} 0, & \text{if } n_i^{\Upsilon}(M_i(s)) < n_i^{\Im}(M_i(s))\\ 1, & \text{if } n_i^{\Im}(M_i(s)) < n_i^{\Upsilon}(M_i(s))\\ \phi, & \text{if } n_i^{\Im}(M_i(s)) = n_i^{\Upsilon}(M_i(s))\end{cases} \quad (6)$$

and

$$n_i(M_i(s)) = \min\{n_i^{\Im}(M_i(s)), n_i^{\Upsilon}(M_i(s))\}. \quad (7)$$



(a)

(b)

Fig. 1. Automata G and R of Example 1.

Example 1: We consider a plant modeled by the finite automaton G shown in Fig. 1(a). Let $|I| = 2$,

$$M_1(\sigma) = \begin{cases} \sigma, & \text{if } \sigma \in \{a, a', c, e\} \\ \varepsilon, & \text{otherwise,} \end{cases}$$
$$M_2(\sigma) = \begin{cases} \sigma, & \text{if } \sigma \in \{b, b', c, e\} \\ \varepsilon, & \text{otherwise.} \end{cases}$$

Also, let $K \subseteq L$ be a language generated by the finite automaton R shown in Fig. 1(b).

We synthesize the decentralized prognoser using (4)–(7) for $N = 2$. We first need to compute the language pairs $\{(\Im_k, \Upsilon_k)\}_{0 \le k \le 2}$. Initially, we have

$$\begin{aligned} \Im_0 &= ac + bc + e(c + ab'(\varepsilon + c) + ba'(\varepsilon + c)), \\ \Upsilon_0 &= pr(cd^* + ab'cd^* + ba'cd^* + e(acd^* + bcd^*)). \end{aligned}$$

Since

$$\begin{aligned} M_1(\Im_0) &= ac + c + e(c + a(\varepsilon + c) + a'(\varepsilon + c)), \\ M_2(\Im_0) &= c + bc + e(c + b'(\varepsilon + c) + b(\varepsilon + c)), \\ M_1(\Upsilon_0) &= pr(c + ac + a'c + e(ac + c)), \\ M_2(\Upsilon_0) &= pr(c + b'c + bc + e(c + bc)), \end{aligned}$$

we have

$$\begin{aligned} \Im_1 &= \Im_0 \cap \left(\bigcap_{i \in I} M_i^{-1} M_i(\Upsilon_0) \right) \\ &= ac + bc + ec, \\ \Upsilon_1 &= \Upsilon_0 \cap \left(\bigcap_{i \in I} M_i^{-1} M_i(\Im_0) \right) \\ &= cd^* + e(acd^* + bcd^*). \end{aligned}$$

Also, since

$$\begin{aligned} M_1(\Im_1) &= ac + c + ec, \\ M_2(\Im_1) &= c + bc + ec, \\ M_1(\Upsilon_1) &= c + e(ac + c), \\ M_2(\Upsilon_1) &= c + e(c + bc), \end{aligned}$$

TABLE I

LOCAL DECISIONS OF P_1 AND P_2.

$t \in M_1(K)$	$n_1^{\Im}(t)$	$n_1^{\Upsilon}(t)$	$c_1(t)$	$n_1(t)$
ε	1	0	0	0
c	3	2	0	2
a	1	0	0	0
ac	1	2	1	1
a'	1	0	0	0
$a'c$	1	0	0	0
e	1	0	0	0
ec	2	3	1	2
ea	1	1	ϕ	1
eac	2	1	0	1
ea'	0	1	1	0
$ea'c$	0	1	1	0
$t \in M_2(K)$	$n_2^{\Im}(t)$	$n_2^{\Upsilon}(t)$	$c_2(t)$	$n_2(t)$
ε	1	0	0	0
c	3	2	0	2
b	1	0	0	0
bc	1	2	1	1
b'	1	0	0	0
$b'c$	1	0	0	0
e	1	0	0	0
ec	2	3	1	2
eb	1	1	ϕ	1
ebc	2	1	0	1
eb'	0	1	1	0
$eb'c$	0	1	1	0

we have

$$\begin{aligned} \Im_2 &= \Im_1 \cap \left(\bigcap_{i \in I} M_i^{-1} M_i(\Upsilon_1) \right) = ec, \\ \Upsilon_2 &= \Upsilon_1 \cap \left(\bigcap_{i \in I} M_i^{-1} M_i(\Im_1) \right) = cd^*. \end{aligned}$$

The local decisions of P_1 and P_2 computed using (4)–(7) are shown in Table I. For example, $P_1(ac)$ is computed as follows. Since $ac \in M_1(\Upsilon_0) - M_1(\Upsilon_1)$, we have by (4) that $n_1^{\Im}(ac) = 1$. Also, since $ac \in M_1(\Im_1) - M_1(\Im_2)$, we have by (5) that $n_1^{\Upsilon}(ac) = 2$. It follows that $1 = n_1^{\Im}(ac) < n_1^{\Upsilon}(ac) = 2$. By (6) and (7), we have $c_1(ac) = 1$ and $n_1(ac) = 1$, which implies that P_1 issues a prognostic decision "1" following the observation $ac \in M_1(K)$ with the ambiguity level 1.

Then, the global prognostic decisions of the decentralized prognoser $\{P_i\}_{i \in I}$ are computed as shown in Table II. For example, $\{P_i\}_{i \in I}(ac)$ is computed as follows. Since $1 = n_1(M_1(ac)) < n_2(M_2(ac)) = 2$ and $c_1(M_1(ac)) = 1$, we have $n(ac) = 1$ and $\{P_i\}_{i \in I}(ac) = 1$.

The following lemma shows that the decentralized prognoser given by (4)–(7) is an N-inferring one with no false alarms.

Lemma 1: [6] Consider the decentralized prognoser $\{P_i\}_{i \in I} : K \to C$ consisting of local prognosers $P_i : M_i(K) \to C \times \mathcal{N}$ $(i \in I)$, and defined by (4)–(7). Then, $\{P_i\}_{i \in I}$ is an N-inferring decentralized prognoser satisfying (2) and (3).

We introduce the notion of N-inference-prognosability and show that the decentralized prognoser also has no missed detections under the decentralized prognosis performed using

TABLE II

GLOBAL DECISIONS OF $\{P_i\}_{i \in I}$.

$s \in K$	$n(s)$	$\{P_i\}_{i\in I}(s)$
ε	0	0
cd^*	2	0
a, b	0	0
ac, bc	1	1
ab', ba'	0	0
$ab'cd^*$, $ba'cd^*$	0	0
e	0	0
ec	2	1
ea, eb	0	0
$eacd^*$, $ebcd^*$	1	0
eab', eba'	0	1
$eab'c$, $eba'c$	0	1

the local and global prognosers given by (4)–(7) under this condition. In fact this condition serves as a necessary and sufficient condition for the existence of an N-inferring decentralized prognoser with no missed detections and false alarms.

Definition 3: The pair (L, K) of closed languages with $K \subseteq L$ is said to be N*-inference-prognosable* if $\partial_L(K) \subseteq \Im_L(K) - \Im_{N+1}$.

The following lemma states that if (L, K) is N-inference-prognosable, then there are no missed detections under the decentralized prognosis performed using the local and global prognosers given by (4)–(7).

Lemma 2: Consider the decentralized prognoser $\{P_i\}_{i\in I} : K \to C$ consisting of local prognosers $P_i : M_i(K) \to C \times \mathcal{N}$ ($i \in I$), and defined by (4)–(7). If the pair (L, K) of closed languages is N-inference-prognosable, then (1) holds.

The following theorem establishes the main result of the paper.

Theorem 1: Given a pair (L, K) of closed languages with $K \subseteq L$, there exists an N-inferring decentralized prognoser $\{P_i\}_{i\in I} : K \to C$ satisfying (1), (2), and (3) if and only if (L, K) is N-inference-prognosable.

In the following we show that the system of Example 1 is 2-inference-prognosable but it is not 1-inference-prognosable.

Example 2: We revisit the setting of Example 1. We have

$$\begin{aligned}\partial_L(K) &= ac + bc + e(c + ab'c + ba'c),\\ \Im_L(K) &= ac + bc + e(c + ab'(\varepsilon + c) + ba'(\varepsilon + c)),\\ \Im_2 &= ec,\\ \Upsilon_2 &= cd^*.\end{aligned}$$

It follows that $\partial_L(K) \not\subseteq \Im_L(K) - \Im_2$, which implies that (L, K) is not 1-inference-prognosable. Since

$$\begin{aligned}M_1(\Im_2) &= M_2(\Im_2) = ec,\\ M_1(\Upsilon_2) &= M_2(\Upsilon_2) = c,\end{aligned}$$

we have

$$\Im_3 = \Im_2 \cap \left(\bigcap_{i\in I} M_i^{-1}M_i(\Upsilon_2)\right) = \emptyset,$$

$$\Upsilon_3 = \Upsilon_2 \cap \left(\bigcap_{i\in I} M_i^{-1}M_i(\Im_2)\right) = \emptyset.$$

We have $\partial_L(K) \subseteq \Im_L(K) - \Im_3$, which implies that (L, K) is 2-inference-prognosable.

By Table II, we can verify that $\{P_i\}_{i\in I}$ is a 2-inferring decentralized prognoser satisfying (1), (2), and (3).

Remark 1: We discuss how to verify N-inference-prognosability of a pair (L, K) of closed regular languages with $K \subseteq L$. The verification of N-inference-prognosability requires checking emptiness of the languages $\partial_L(K) \cap (\Sigma^* - \Im_L(K))$ and $\partial_L(K) \cap \Im_{N+1}$. Since checking emptiness of a language can be done linearly in the size of an acceptor of the language, we only discuss the computation of certain acceptors for $\partial_L(K) \cap (\Sigma^* - \Im_L(K))$ and $\partial_L(K) \cap \Im_{N+1}$. A point of our construction is to establish that the step of "determinization" (which is exponential in the size of the automaton being determinized) is never required.

Let $G = (X, \Sigma, \alpha, X_0, X)$ be the finite plant model with $L(G) = L_m(G) = L$, and $R = (Y, \Sigma, \beta, Y_0, Y)$ be a finite deterministic generator of the nonfailure specification language, i.e., $L(R) = L_m(R) = K$. For computing $\partial_L(K)$, we construct the synchronous composition $G\|R := (Z, \Sigma, \gamma, Z_0, Z)$ of G and R, where $Z = X \times Y$, $Z_0 = X_0 \times Y_0$, and $\gamma : Z \times (\Sigma \cup \{\varepsilon\}) \to 2^Z$ is defined in the usual manner. Then $L(G\|R) = L(G) \cap L(R) = K$ holds. Let $Z^\partial := \{(x, y) \in Z \mid \exists\sigma \in \Sigma : \ \alpha(x, \sigma) \neq \emptyset, \beta(y, \sigma) = \emptyset\}$. Then, for the finite automaton $(G\|R)^\partial := (Z, \Sigma, \gamma, Z_0, Z^\partial)$, we have $L_m((G\|R)^\partial) = \partial_L(K)$.

Next we discuss the computation of $\Im_k$ and Υ_k ($k \geq 0$) inductively over k. For the base step ($k = 0$), let $\Im(Y) \subseteq Y$ be the set of *indicator* states of R from which no cycle in R can be reached [8]. Also let $\Upsilon(Y) \subseteq Y$ be the set of *nonindicator* states of R from which a cycle in R can be reached [8]. The identification of $\Im(Y)$ and $\Upsilon(Y)$ can be performed in complexity linear in the size of R. Then the languages $\Im_L(K)(= \Im_0)$ and $\Upsilon_L(K)(= \Upsilon_0)$ are computed by replacing the marked state set Y of R with $\Im(Y)$ and $\Upsilon(Y)$, respectively. Note to check the emptiness of $\partial_L(K) \cap (\Sigma^* - \Im_L(K))$, the complement $\Sigma^* - \Im_L(K)$ of $\Im_L(K)$ can be computed in the usual manner, and the intersection $\partial_L(K) \cap (\Sigma^* - \Im_L(K))$ can be computed using the synchronous composition operator.

We discussed above the computation of $\Im_0$ and Υ_0 (the base step). For the induction step, let $R_{\Im_k}$ and R_{Υ_k} be finite acceptors of $\Im_k$ and Υ_k, respectively. For each $i \in I$, a finite acceptor of $M_i^{-1}M_i(\Im_k)$ is constructed as follows: Replace each transition that exists in $R_{\Im_k}$ by a set of transitions on all M_i-indistinguishable events (including ε). Note that since an ε-transition is implicitly defined at each state as a self-loop, unobservable events will get added as self-loops at each state of $R_{\Im_k}$. Then, the resulting, possibly nondeterministic, automaton accepts $M_i^{-1}M_i(\Im_k)$. It should be noted that this resulting automaton, denoted by $M_i^{-1}M_i(R_{\Im_k})$, has the same state set as $R_{\Im_k}$. In the same way, we can construct a finite automaton accepting $M_i^{-1}M_i(\Upsilon_k)$, denoted by $M_i^{-1}M_i(R_{\Upsilon_k})$. Then, the synchronous compositions $R_{\Im_{k+1}} := R_{\Im_k} \| (\|_{i \in I} M_i^{-1} M_i(R_{\Upsilon_k}))$ and $R_{\Upsilon_{k+1}} := R_{\Upsilon_k} \| (\|_{i \in I} M_i^{-1} M_i(R_{\Im_k}))$ accept $\Im_{k+1}$ and Υ_{k+1}, respectively. Let $Y_{\Im_k}$ and Y_{Υ_k} be the state sets of $R_{\Im_k}$ and R_{Υ_k}, respectively. Then the size of the state sets of $R_{\Im_{k+1}}$ and $R_{\Upsilon_{k+1}}$ are $O(|Y_{\Im_k}| \cdot |Y_{\Upsilon_k}|^{|I|})$ and $O(|Y_{\Upsilon_k}| \cdot |Y_{\Im_k}|^{|I|})$, respectively. Once an acceptor for $\Im_{N+1}$ has been computed (using the above inductive procedure), the emptiness of $\partial_L(K) \cap \Im_{N+1}$ can be checked by a reachability analysis over $(G\|R)^\partial \| R_{\Im_{N+1}}$.

V. Properties of N-Inference-Prognosability

In this section, we study properties of N-inference-prognosable systems. First, we show that the class of coprognosable systems studied in [8] is equivalent to the class of 0-inference-prognosable systems.

Definition 4: [8] A pair (L, K) of closed languages with $K \subseteq L$ is said to be *coprognosable* if

$$(\forall s \in L - K)(\exists t \in pr(s) \cap K)(\exists i \in I)(E_i(t) \subseteq \Im_L(K)),$$

where $E_i(t) := M_i^{-1} M_i(t) \cap K$.

Theorem 2: A pair (L, K) of closed languages with $K \subseteq L$ is 0-inference-prognosable if and only if it is coprognosable.

We also establish that the classes of N-inference-prognosable systems form a monotonically increasing sequence as a function of N. Since the sequence $\{(\Im_k, \Upsilon_k)\}_{k \geq 0}$ of language pairs is monotonically decreasing, the following result is easily obtained.

Theorem 3: For any $N \in \mathcal{N}$, if a pair (L, K) of closed languages with $K \subseteq L$ is N-inference-prognosable, then it is $(N+1)$-inference-prognosable.

The converse relation of Theorem 3 need not hold. For example, the system of Example 1 is 2-inference-prognosable, but not 1-inference-prognosable.

VI. Conclusion

It is desirable to have systems in which it is possible to prognose (predict) failures prior to their occurrence. We studied the prognosis of failures in a decentralized framework where multiple prognosers, based on their observations of the executed behavior, infer the inevitability of an impending failure. The decentralized set of prognosers use a certain inferencing mechanism, that was originally introduced in the setting of control [7] and also applied in the context of diagnosis [6], [11], to resolve the ambiguities of the self and the others. Each local prognostic decision (of whether or not failure is inevitable) is tagged with an ambiguity level (zero being the minimum), and the global prognostic decision is taken to be the winning local one (i.e., one with the minimum level of ambiguity). We introduced the notion of N-inference-prognosability to characterize the class of systems for which any failure can be predicted prior to its occurrence in a manner that the maximum ambiguity level of a winning decision does not exceed N. An algorithm for verifying N-inference-prognosability is presented. We also showed that a larger N corresponds to a larger class of prognosable systems. Further the class of 0-inference-prognosable systems coincides with the class of coprognosable systems presented in [8], implying that even the class of 1-inference-prognosable systems subsumes the class of coprognosable ones studied in [8].

The inferencing-based decentralized decision-making framework used here can also be cast in the modal logic framework of [2], where the underlying Kripke structure will have as "possible worlds", the set of all nonfailure specification traces, and an i-labeled edge will exist between a pair of possible worlds if and only if they are indistinguishable to the site-i prognoser. Note, unlike the setting of [2] where the number of possible worlds is taken to be finite, the number of possible worlds in our setting can be infinite since the nonfailure specification language can contain infinitely many traces, and each such trace constitutes one possible world. Thus the existing results of modal logic cannot directly be applied, and the results presented in the paper can be viewed to supplement the existing results of modal logic.

References

[1] H. K. Fadel and L. E. Holloway, "Using SPC and template monitoring method for fault detection and prediction in discrete event manufacturing systems," in *Proc. 1999 IEEE International Symposium on Intelligent Control/Intelligent Systems and Semiotics*, Cambridge, MA, pp. 150–155, 1999.

[2] R. Fagin, J. Y. Halpern, Y. Moses, and M . Y. Vardi, *Reasoning About Knowledge*. Cambridge, MA: The MIT Press, 1995.

[3] S. Genc and S. Lafortune, "Predictability in discrete-event systems under partial observation," in *Preprints of IFAC Symposium on Fault Detection, Supervision, and Safety of Technical Processes*, Beijing, China, 2006.

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

[5] S. Jiang and R. Kumar, "Diagnosis of repeated failures for discrete event systems with linear-time temporal-logic specifications," *IEEE Trans. Autom. Sci. Eng.*, vol. 3, no. 1, pp. 47–59, 2006.

[6] R. Kumar and S. Takai, "Inference-based ambiguity management in decentralized decision-making: Decentralized diagnosis of discrete event systems," in *Proc. 2006 Amer. Control Conf.*, Minneapolis, MN, pp. 6069–6074.

[7] R. Kumar and S. Takai, "Inference-based ambiguity management in decentralized decision-making: Decentralized control of discrete event systems," *IEEE Trans. Autom. Control*, vol. 52, no. 10, pp. 1783–1794, 2007.

[8] R. Kumar and S. Takai. "Decentralized prognosis of failures in discrete event systems," in *Proc. 9th Int. Workshop Discrete Event Systems*, Göteborg, Sweden, pp. 376–381, 2008.

[9] S. L. Ricker and K. Rudie, "Know means no: Incorporating knowledge into discrete-event control systems," *IEEE Trans. Autom. Control*, vol. 45, no. 9, pp. 1656–1668, 2000.

[10] S. L. Ricker and K. Rudie, "Knowledge is a terrible thing to waste: Using inference in discrete-event control problems," *IEEE Trans. Autom. Control*, vol. 52, no. 3, pp. 428–441, 2007.

[11] S. Takai and R. Kumar, "Decentralized diagnosis for nonfailures of discrete event systems using inference-based ambiguity management," in *Proc. 8th Int. Workshop Discrete Event Systems*, Ann Arbor, MI, pp. 242–247, 2006.

[12] G. Vachtsevanos, F. L. Lewis, M. Roemer, A. Hess, and B. Wu, *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*. Hoboken, NJ: John Wiley and Sons, 2006.

[13] T.-S. Yoo and S. Lafortune, "Decentralized supervisory control with conditional decisions: Supervisor existence," *IEEE Trans. Autom. Control*, vol. 49, no. 11, pp. 1886–1904, 2004.