Reliable Decentralized Fault Prognosis of Discrete-Event Systems

Xiang Yin and Zhaojian Li

***Abstract*—We investigate the problem of reliable decentralized fault prognosis of partially-observed discrete-event systems. In this problem, n local prognosers are deployed to send their local prognostic decisions to a coordinator that calculates the final prognostic decision. However, only k ($1 \leq k \leq n$) local prognostic decisions are guaranteed to be available to the coordinator due to possible failures or communication losses of at most $n - k$ local prognosers. We propose the notion of k-reliable decentralized prognoser in order to address this reliability issue. A necessary and sufficient condition for the existence of a k-reliable decentralized prognoser, which predicts faults prior to their occurrences, is presented. This condition is termed as k-reliable coprognosability. A polynomial-time algorithm for the verification of k-reliable coprognosability is presented. We also demonstrate how to compute the k-reliable reactive bound prior to any occurrence of faults.**

***Index Terms*—Decentralized fault prognosis, discrete-event systems, reliable coprognosability, reliable prognosers.**

I. Introduction

Fault prognosis is an important task for safety critical systems. In many complex large-scale systems, the information structure is decentralized. In this correspondence paper, we consider the decentralized fault prognosis problem of discrete event systems (DESs). In this problem, the plant is monitored by a set of local agents (or local prognosers) that make local prognostic decisions based on their own observations. The local prognostic decisions are sent to the fusion site (or the coordinator) in order to calculate a global prognostic decision. The goal is to predict any fault of the system prior to its occurrence with no missed alarm and no false alarm.

The problem of fault prognosis has recently attracted considerable attention in the DES literature (see [4]–[6], [8]–[13], [18], [23]–[25], [29]). Fault prognosis of DESs was initialed in [8], where the notion of uniformly bounded prognosability, which is termed as predictability in [8], was proposed. Roughly speaking, a DES is prognosable if: 1) any fault can be alarmed prior to its occurrence, i.e., no missed alarm and 2) once an alarm is issued, a fault will occur for sure within a finite number of steps, i.e., no false alarm. In [12], the problem of fault prognosis was extended to the decentralized setting and the notions of coprognosability and uniformly bounded coprognosability were introduced. It was shown that these two notions are equivalent when the languages under consideration are regular. In [10], [11], and [24], decentralized fault prognosis under different architectures was studied. More recently, the problem of fault progosis of DES was further investigated for distributed systems [25], [29], timed systems [4], stochastic systems [5], [6], [18], and systems modeled by Petri-nets [13].

Previous studies on decentralized fault prognosis rely on the implicit assumption that the system is reliable in the sense that the coordinator always has access to all local prognostic decisions. However, this assumption may not hold in many systems due to the following reasons. First, as depicted in Fig. 1, in many distributed system, the local decisions are sent to the coordinator via a communication network in which communication delays or signal losses may occur [14], [22]. Therefore, one must take these issues into account. Second, each local prognoser itself may not be reliable due to local hardware or software errors. In order to evaluate the robustness or the reliability of the system, many different approaches were proposed in [1], [2], [14]–[17], [19]–[22], and [26]–[28]. In particular, in [16] and [26], the problem of reliable decentralized supervisory control was studied by assuming that certain amount of local control decisions may be lost. This approach was also extended to the decentralized fault diagnosis problem in [1], [17], and [28].

Motivated by the reliable control and diagnosis techniques mentioned above, in this paper, we consider the problem of reliable decentralized fault prognosis. More specifically, we assume that there are n local prognosers sending their local prognostic decisions to the coordinator, but the coordinator is only guaranteed to receive k local decisions, where $k \leq n$ is a non-negative integer. In other words, we consider the scenario in which at most $n - k$ local prognostic decisions may be lost. The contributions of this paper are as follows. First, we provide the necessary and sufficient condition, termed as k-reliable coprognosability, for the existence of a k-reliable decentralized prognoser. Second, we develop a polynomial-time algorithm for the verification of k-reliable coprognosability. We show that, when using the approach proposed, the verification of k-reliable coprognosability is not more computationally difficult than the verification of coprognosability under reliable decisions. Furthermore, when the system is k-reliably coprognosable, a method for the computation of k-reliable reactive bound is presented. The k-reliable reactive bound quantitatively specifies the amount of time the prognoser can react prior to the occurrence of a fault.

In [23], the problem of robust fault prognosis was studied. Our paper is clearly different from the above mentioned work due to the following reasons. First, we study the decentralized prognosis problem while [23] considers the centralized setting. Second, the robustness in [23] is defined in the sense that there are a set of possible models of the system, which is also clearly different from the notion of reliability investigated in this paper.

The remainder of this paper is organized as follows. In Section II, we describe some preliminaries and notations used in this paper. In Section III, the problem of reliable decentralized prognosis is formulated and the notion of k-reliable coprognosability is presented. Section IV provides a polynomial-time algorithm for the verification of k-reliable coprognosability. The computation of k-reliable reactive bound is studied in Section V. Finally, we conclude the paper in Section VI.

II. Preliminaries

A. System Model

We first review some common notations of DES; the reader is referred to [3] for more details. Let Σ^* be the set of all finite strings over a finite set of events Σ, including the empty string ϵ. A language

Manuscript received June 28, 2015; accepted September 26, 2015. Date of publication November 26, 2015; date of current version October 13, 2016. This paper was recommended by Associate Editor I. Makki.

X. Yin is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: xiangyin@umich.edu).

Z. Li is with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: zhaojli@umich.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2015.2499178

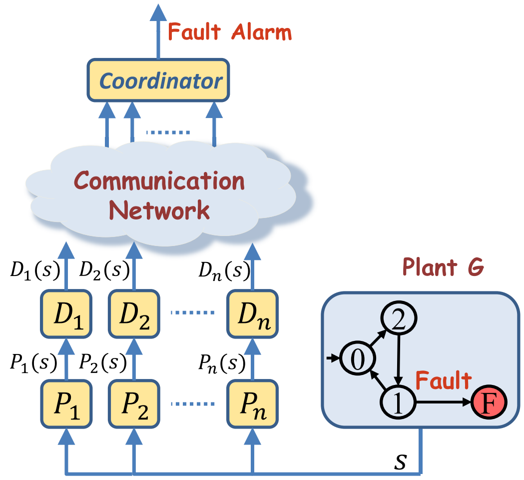

Fig. 1. Decentralized fault prognosis architecture.

$L \subseteq \Sigma^*$ is a subset of Σ^* and $\overline{L}$ is the prefix-closure of language L defined by $\overline{L} = \{t \in \Sigma^* : \exists u \in \Sigma^* \text{ s.t. } tu \in L\}$. Given a language L and a string $s \in L$, we denote by L/s the post-language of s, i.e., $L/s := \{t \in \Sigma^* : st \in L\}$.

A DES is modeled by a deterministic finite-state automaton $G = (Q, \Sigma, \delta, q_0, Q_m)$, where Q is the finite set of states, Σ is the finite set of events, $\delta : Q \times \Sigma \to Q$ is the partial transition function, $q_0 \in Q$ is the initial state and Q_m is the set of marked states. We denote by $G = (Q, \Sigma, \delta, q_0)$ an automaton if marking is not considered. The transition function δ is extended to $Q \times \Sigma^*$ in the usual manner (see [3]). For brevity, we write $\delta(q_0, s)$ as $\delta(s)$, i.e., $\delta(s)$ is the state reached via s from the initial state. The language generated by G from state q is defined by $\mathcal{L}(G, q) = \{s \in \Sigma^* : f(q, s)!\}$, where ! means "is defined." We write $\mathcal{L}(G, q)$ as $\mathcal{L}(G)$ when $q = q_0$, i.e., $\mathcal{L}(G)$ is the language generated by G. The language marked by G is $\mathcal{L}_m(G) = \{s \in \Sigma^* : \delta(q_0, s) \in Q_m\}$. The system G is said to be live if $\forall q \in Q, \exists \sigma \in \Sigma : \delta(q, \sigma)!$. Hereafter, we assume that G is live. This assumption is without loss of generality (w.l.o.g.), since we can add self-loop at each state in G from which no transition is defined.

Given two automata $G = (Q, \Sigma, \delta, q_0)$ and $H = (Q_H, \Sigma, \delta_H, q_{0,H})$, we say that H is a sub-automaton of G, denoted by $H \sqsubseteq G$, if: 1) $q_{0,H} = q_0$; 2) $Q_H \subseteq Q$; and 3) for any $q \in Q_H, \sigma \in \Sigma$, we have that $\delta_H(q, \sigma) = \delta(q, \sigma)$ if $\delta_H(q, \sigma)!$. We say that H is a strict sub-automaton of G, denoted by $H \sqsubset G$, if: 1) $H \sqsubseteq G$ and 2) $\forall s \in \mathcal{L}(G) \setminus \mathcal{L}(H) : \delta(q_0, s) \notin Q_H$. In other words, strict sub-automaton requires that once a string leaves the state space of H, it should never go back. It was shown in [7] that if $\mathcal{L}(H) \subseteq \mathcal{L}(G)$, then we can always construct two new automata G' and H' such that 1) $H' \sqsubset G'$ and 2) $\mathcal{L}(H) = \mathcal{L}(H')$ and $\mathcal{L}(G) = \mathcal{L}(G')$. Therefore, we can always assume w.l.o.g. that $H \sqsubset G$ when $\mathcal{L}(H) \subseteq \mathcal{L}(G)$.

B. Decentralized Fault Prognosis

Let G be the system automaton and H be the specification automaton, i.e., $\mathcal{L}(H)$ is a nonempty prefix-closed specification language that represents the normal behavior of the system. Then strings in $\mathcal{L}(G) \setminus \mathcal{L}(H)$ are considered as fault behaviors and we want to predict the occurrence of any string in $\mathcal{L}(G) \setminus \mathcal{L}(H)$. As we mentioned earlier, hereafter, we assume w.l.o.g. that the specification automaton $H = (Q_H, \Sigma, \delta_H, q_{0,H})$ is a strict sub-automaton of the system automaton $G = (Q, \Sigma, \delta, q_0)$, i.e., $H \sqsubset G$. Therefore, under this assumption, we know that string $s \in \mathcal{L}(G)$ is a nonfault string if and only $\delta(s) \in Q_H$ and string $s \in \mathcal{L}(G)$ is a fault string if and only $\delta(s) \in Q \setminus Q_H$.

In the decentralized fault prognosis framework [12], the system is monitored by a set of agents (or local prognosers). We assume that there are n local prognosers and we denote by $\mathcal{I} = \{1, \ldots, n\}$ the index set. For each prognoser $i \in \mathcal{I}$, $\Sigma_{o,i}$ is the set of events it can observe. Also, for any $i \in \mathcal{I}$, $P_i : \Sigma^* \to \Sigma_{o,i}^*$ is the natural projection defined by

$$P_i(\epsilon) = \epsilon \quad P_i(s\sigma) = \begin{cases} P_i(s)\sigma & \text{if } \sigma \in \Sigma_{o,i} \\ P_i(s) & \text{if } \sigma \notin \Sigma_{o,i}. \end{cases} \tag{1}$$

Each local prognoser $i \in \mathcal{I}$ is defined as a function $D_i : P_i(\mathcal{L}(H)) \to \{0, 1\}$, where "1" means a fault will occur in a finite number of steps, i.e., a fault is inevitable, and "0" means a fault is not guaranteed to occur within a finite number of steps. Each local prognoser sends its local prognostic decision to a coordinator in order to calculate a global prognostic decision. In this paper, we consider the disjunctive architecture, i.e., the global decision is "1" if one local prognoser says "1." Therefore, under this architecture, the decentralized prognoser is the function $\{D_i\}_{i \in \mathcal{I}} : \mathcal{L}(H) \to \{0, 1\}$ defined as follows. For any $s \in \mathcal{L}(H)$

$$\{D_i\}_{i \in \mathcal{I}}(s) = 1 \Leftrightarrow \exists i \in \mathcal{I} : D_i(P_i(s)) = 1. \tag{2}$$

In [12], two criteria, no missed alarm and no false alarm, were proposed in order to evaluate the validation of a decentralized prognoser.

Definition 1: A decentralized prognoser $\{D_i\}_{i \in \mathcal{I}}$ is said to be valid if the following two properties hold.

1) No missed alarm, that is

$$(\forall s \in \mathcal{L}(G) \setminus \mathcal{L}(H))\big(\exists t \in \overline{\{s\}} \cap \mathcal{L}(H)\big)\big[\{D_i\}_{i \in \mathcal{I}}(t) = 1\big]. \tag{3}$$

2) No false alarm, that is

$$\forall s \in \mathcal{L}(H) : \big[\{D_i\}_{i \in \mathcal{I}}(s) = 1\big] \Rightarrow (\exists m \in \mathbb{N})(\forall t \in \mathcal{L}(G)/s)\big[|t| \geq m \Rightarrow st \in \mathcal{L}(G) \setminus \mathcal{L}(H)\big]. \tag{4}$$

It was shown in [12] that the notion of coprognosability, whose definition is recalled next, provides the necessary and sufficient condition for the existence of a valid decentralized prognoser that predicts faults with no error.

Definition 2 (Coprognosability): A specification $\mathcal{L}(H)$ is said to be coprognosable with respect to $\mathcal{L}(G)$ and $\Sigma_{o,i}, i \in \mathcal{I}$ if

$$\begin{aligned} &(\exists m \in \mathbb{N})(\forall s \in \mathcal{L}(G) \setminus \mathcal{L}(H))\big(\exists t \in \overline{\{s\}} \cap \mathcal{L}(H)\big) \\ &(\exists i \in \mathcal{I})\Big(\forall u \in P_i^{-1}P_i(t) \cap \mathcal{L}(H)\Big)(\forall v \in \mathcal{L}(G)/u) \\ &\big[|v| \geq m \Rightarrow uv \in \mathcal{L}(G) \setminus \mathcal{L}(H)\big]. \end{aligned} \tag{5}$$

III. Reliable Decentralized Fault Prognosis

In this section, we first introduce the notion of k-reliable prognoser. Then we present the necessary and sufficient condition, under which there exists a k-reliable prognoser.

First, we note that, in the standard decentralized fault prognosis framework, it is assumed that the decentralized system is reliable in the sense that the coordinator can always receive all the local prognostic decisions. However, as we discussed earlier, this assumption may not hold due to communication losses or errors at the local sites. Therefore, in order to take this reliability issue into account, hereafter, we assume that there are only k local prognostic decisions available to the coordinator at each step, where $1 \leq k \leq n$. In other words, at most $n - k$ local prognostic decisions may be lost. We first propose the notion of k-reliable prognoser that can predict the occurrences of faults even if only k local prognostic decisions are available to the coordinator at each time.

Definition 3 (k-Reliable Prognoser): A valid decentralized prognoser $\{D_i\}_{i\in\mathcal{I}}$ is said to be k-reliable if

$$(\forall s \in \mathcal{L}(G)\setminus \mathcal{L}(H))(\exists t \in \overline{\{s\}} \cap \mathcal{L}(H))$$
$$[|\{i \in \mathcal{I} : D_i(P_i(s)) = 1\}| \geq n - k + 1]. \quad (6)$$

Before we present the necessary and sufficient condition for the existence of a k-reliable decentralized prognoser, we first introduce some necessary notions. In [12], the notions of boundary strings, indicator strings and nonindicator strings were introduced. Since we have assumed w.l.o.g. that H is a strict sub-automaton of G, we define three similar notions, in terms of states rather than strings, as follows.

Definition 4: Let H be the specification automaton and G be the system automaton with $H \sqsubset G$.

1) A boundary state is a state in H, from which an event that violates the specification can occur. We denote by $\partial(H, G)$ the set of boundary states, i.e., $\partial(H, G) = \{q \in Q_H : \exists\sigma \in \Sigma \text{ s.t. } \delta(q, \sigma)! \wedge \delta_H(q, \sigma)/!\}$.
2) An indicator state is a state in H, from which a string that violates the specification will occur for sure within a finite number of steps. We denote by $\Im(H, G)$ the set of indicator states, i.e., $\Im(H, G) = \{q \in Q_H : \exists m \in \mathbb{N}, \forall t \in \mathcal{L}(G, q) \text{ s.t. } |t| \geq m \wedge \delta(q, t) \in Q \setminus Q_H\}$.
3) A nonindicator state is a state in H, from which an arbitrarily long nonfault behavior can occur, i.e., $\Upsilon(H, G) = Q_H \setminus \Im(H, G) = \{q \in Q_H : \forall m \in \mathbb{N}, \exists t \in \Sigma^* \text{ s.t. } |t| \geq m \wedge \delta_H(q, t)!\}$.

For each local prognoser $i \in \mathcal{I}$, we also denote by $\mathcal{E}_i^H(s)$ agent i's state estimate of s under $\Sigma_{o,i}$ with respect to the state space of H, that is

$$\mathcal{E}_i^H(s) := \{q \in Q_H : \exists t \in \mathcal{L}(H) \text{ s.t. } P_i(s) = P_i(t) \wedge \delta_H(t) = q\}. \quad (7)$$

Then for any string $s \in \mathcal{L}(H)$, we denote by $\mathcal{I}_D(s)$ the set of agents whose state estimates only consist of indicator states, that is

$$\mathcal{I}_D(s) := \left\{i \in \mathcal{I} : \mathcal{E}_i^H(s) \subseteq \Im(H, G)\right\}. \quad (8)$$

With the notions introduced above, we define the notion of k-reliable coprognosability as follows.

Definition 5 (k-Reliable Coprognosability): A specification $\mathcal{L}(H)$ is said to be k-reliably coprognosable with respect to $\mathcal{L}(G)$ and $\Sigma_{o,i}, i \in \mathcal{I}$ if

$$(\forall s \in \mathcal{L}(G) \setminus \mathcal{L}(H))\big(\exists t \in \overline{\{s\}} \cap \mathcal{L}(H)\big)[|\mathcal{I}_D(t)| \geq n - k + 1]. \quad (9)$$

The following result reveals that k-reliable coprognosability provides the necessary and sufficient condition for the existence of a valid k-reliable decentralized prognoser.

Theorem 1 (Existence Condition): There exists a valid k-reliable decentralized prognoser $\{D_i\}_{i\in\mathcal{I}}$, if and only if, $\mathcal{L}(H)$ is k-reliably coprognosable with respect to $\mathcal{L}(G)$ and $\Sigma_{o,i}, i \in \mathcal{I}$.

Proof: ($\Leftarrow$) By construction. Let us consider a decentralized prognoser $\{D_i\}_{i\in\mathcal{I}}$ defined as follows. For each local prognoser $i \in \mathcal{I}$, for any string $s \in \mathcal{L}(H)$, we have

$$D_i(P_i(s)) = 1 \Leftrightarrow \mathcal{E}_i^H(s) \subseteq \Im(H, G). \quad (10)$$

Under the above prognostic strategy, we know that $|\mathcal{I}_D(s)| = |\{i \in \mathcal{I} : D_i(P_i(s)) = 1\}|$. Therefore, the condition that $\mathcal{L}(H)$ is k-reliably coprognosable, i.e., (9) holds, implies that (3) and (6) hold for $\{D_i\}_{i\in\mathcal{I}}$. Next, we show by contradiction that (4) holds. Assume that (4) does not hold, we know that $(\exists s \in \mathcal{L}(H))(\exists i \in \mathcal{I} : D_i(P_i(s)) = 1)(\forall m \in \mathbb{N})(\exists t \in \mathcal{L}(G)/s)[|t| \geq m \wedge st \in \mathcal{L}(H)]$. First, we know that $\delta_H(s) \in \Upsilon(H, G)$, since $(\forall m \in \mathbb{N})(\exists t \in \mathcal{L}(G)/s)[|t| \geq m \wedge st \in \mathcal{L}(H)]$. Second, by $D_i(P_i(s)) = 1$, we know that $\delta_H(s) \in \mathcal{E}_i^H(s) \subseteq \Im(H, G)$. However, this is a contradiction since sets $\Upsilon(H, G)$ and $\Im(H, G)$ are disjoint. Therefore, (4) holds. Overall, we know that $\{D_i\}_{i\in\mathcal{I}}$ is a valid k-reliable decentralized prognoser.

($\Rightarrow$) Consider a string $s \in \mathcal{L}(G)\setminus\mathcal{L}(H)$. By (6), we know that there exists $t \in \overline{\{s\}} \cap \mathcal{L}(H)$ such that $|\{i \in \mathcal{I} : D_i(P_i(s)) = 1\}| \geq n - k + 1$. Let $\mathcal{P} = \{i_1, \ldots, i_{|\mathcal{P}|}\}, |\mathcal{P}| \geq n - k + 1$ be the set of agents whose decisions are "1." By (4), we know that $\forall s \in \mathcal{L}(H) : \{D_i\}_{i\in\mathcal{I}}(s) = 1 \Rightarrow \delta_H(s) \in \Im(H, G)$. Then, for any agent $i \in \mathcal{P}$, since $\forall t \in P_i^{-1}(P_i(s)) : D_i(P_i(t)) = D_i(P_i(s)) = 1$, which means $\{D_i\}_{i\in\mathcal{I}}(t) = 1$, we know that $\delta_H(t) \in \Im(H, G)$. Therefore, we know that for any $i \in \mathcal{P}$, $\mathcal{E}_i^H(s) \subseteq \Im(H, G)$, i.e., $|\mathcal{I}_D(s)| \geq |\mathcal{P}| \geq n - k + 1$. Hence, $\mathcal{L}(H)$ is k-reliably coprognosable. ■

Remark 1: It is easy to verify that, by using the notions of boundary state and indicator state, (5) can be rewritten as $(\forall s \in \mathcal{L}(G)\setminus\mathcal{L}(H))(\exists t \in \overline{\{s\}}\cap\mathcal{L}(H))[|\mathcal{I}_D(t)| \geq 1]$. Hence, by taking $n = k$, k-reliable coprognosability reduces to coprognosability. Therefore, Theorem 1 generalizes the results in [12] to the case where $n - k$ local prognostic decisions are not unreliable.

Clearly, we see that $\mathcal{L}(H)$ k-reliably coprognosable implies that $\mathcal{L}(H)$ is coprognosable. However, the converse relation needs not hold, which is illustrated by the following example.

Example 1: Consider the system automaton G and the specification automaton H in Fig. 2. We have that $\partial(H, G) = \{4\}, \Im = \{2, 3, 4\}$, and $\Upsilon = \{1\}$. Suppose that there are two local prognosers with observable events $\Sigma_{o,1} = \{a, b, o\}$ and $\Sigma_{o,2} = \{a, o\}$, respectively. Clearly, we see that $\mathcal{L}(H)$ is coprognosable and a valid decentralized prognoser $\{D_i\}_{i\in\{1,2\}}$ is defined as follows:

$$D_1(s) = \begin{cases} 1 \text{ if } s \in \left\{o^*a, o^*ab, o^*b, o^*bo\right\} \\ 0 \text{ otherwise} \end{cases} \quad (11)$$

$$D_2(s) = \begin{cases} 1 \text{ if } s \in \left\{o^*a\right\} \\ 0 \text{ otherwise.} \end{cases} \quad (12)$$

However, $\{D_i\}_{i\in\{1,2\}}$ is not a one-reliable prognoser. For example, for string bo, if the agent 1s decision, i.e., $D_1(P_1(bo)) = 1$, is lost, then the global decision is "0," since $D_2(P_2(bo)) = 0$. Therefore, no alarm is provided before the occurrence of fault string bof.

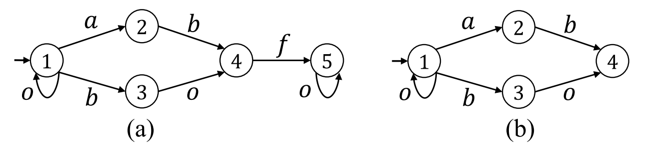

Fig. 2. Specification is coprognosable but not one-reliably coprognosable. (a) G. (b) H.

IV. Verification of Reliable Coprognosability

In this section, we present a polynomial-time algorithm for the verification of k-reliable coprognosability by constructing a new automaton called the k-verifier.

Let $G = (Q, \Sigma, \delta, q_0)$ be the system automaton, $H = (Q_H, \Sigma, \delta_H, \delta_{0,H})$ be the specification automaton, where $H \sqsubset G$, $\Sigma_{o,i}$, and $i \in \mathcal{I}$ be the set of locally observable events. The k-verifier V is defined as the deterministic finite state automaton

$$V = \big(X_V, \Sigma_V, f_V, x_{0,V}, X_{m,V}\big) \quad (13)$$

where

1) $X_V = (\underbrace{Q_H \times \cdots \times Q_H}_{(n+1) \text{ times}}) \cup \{\text{Dead}\}$ is the set of states;
2) $\Sigma_V = \underbrace{(\Sigma \cup \{\epsilon\}) \times \cdots \times (\Sigma \cup \{\epsilon\})}_{(n+1) \text{ times}}$ is the set of events;
3) $x_{0,V} = (\underbrace{q_0, \ldots, q_0}_{n+1 \text{ times}})$ is the initial state;

4) $f_V : X_V \times \Sigma_V \to X_V$ is the partial (deterministic) transition function defined below. For any $\sigma \in \Sigma$, we have the following two types of transitions as follows.
   a) The single transition

$$f_V((q, q_1, \dots, q_n), (\sigma, e_1, \dots, e_n)) = (\delta_H(q, \sigma), \delta_H(q_1, e_1), \dots, \delta_H(q_n, e_n)) \quad (14)$$

   where $\forall i \in \mathcal{I}, e_i = \begin{cases} \sigma, & \text{if } P_i(\sigma) \neq \epsilon \\ \epsilon, & \text{if } P_i(\sigma) = \epsilon \end{cases}$.
   b) The following transition for each $i \in \mathcal{I}$ s.t. $P_i(\sigma) = \epsilon$:

$$f_V\left((q, q_1, \dots, q_n), \left(\epsilon, \epsilon, \dots, \epsilon, \underset{(i+1)^{\text{th}}}{\sigma}, \epsilon, \dots, \epsilon\right)\right) = \left(q, q_1, \dots, q_{i-1}, \delta_H(q_i, \sigma), q_{i+1} \dots, q_n\right). \quad (15)$$

5) $X_{m,V}$ is the set of marked states defined by

$$X_{m,V} := \{(q, q_1, \dots, q_n) \in X_V : q \in \partial(H, G) \wedge |\{i \in \mathcal{I} : q_i \in \Im(H, G)\}| < n - k + 1\}. \quad (16)$$

This completes the definition of the k-verifier.

Remark 2: Let $s = (s_0, s_1, \dots, s_n) \in \mathcal{L}(V)$ be a string in V. Intuitively, the k-verifier tracks one string s_0 representing the system's execution and n strings $s_1, \dots, s_n$ that look identical for agents $1, \dots, n$ under their own observations. Formally, for any string $s = (s_0, s_1, \dots, s_n) \in \mathcal{L}(V)$, we have that $P_i(s_0) = P_i(s_i), \forall i \in \mathcal{I}$. Conversely, for strings $s_0, s_1, \dots, s_n \in \mathcal{L}(H)$, if $P_i(s_0) = P_i(s_i)$, $\forall i \in \mathcal{I}$, then string $(s_0, s_1, \dots, s_n)$ is defined in V from the initial state, i.e., state $(\delta_H(s_0), \delta_H(s_1), \dots, \delta_H(s_n))$ is reachable in V.

The following result states that the specification is not k-reliably coprognosable if and only if a marked state is reached from the initial state in V.

Theorem 2: A specification language $\mathcal{L}(H)$ is k-reliably coprognosable [with respect to $\mathcal{L}(G)$ and $\Sigma_{o,i}, i \in \mathcal{I}$], if and only if, $\mathcal{L}_m(V) = \emptyset$.

Proof: ($\Rightarrow$) By contraposition. Suppose that $\mathcal{L}_m(V) \neq \emptyset$. We know that there exists a string $s = (s_0, s_1, \dots, s_n) \in \mathcal{L}(V)$ such that $f_V(x_{0,V}, s) = (q, q_1, \dots, q_n) \in X_{m,V}$, where $q = \delta_H(s_0)$ and $q_i = \delta_H(s_i), \forall i \in \mathcal{I}$. We define $\mathcal{P} := \{i \in \mathcal{I} : q_i \in \Upsilon(H, G)\}$. By the definition of $X_{m,V}$, we know that $q \in \partial(H, G)$ and $|\mathcal{P}| \geq k$. By the construction of V, we know that $\forall i \in \mathcal{I} : P_i(s_0) = P_i(s_i)$. Therefore, we know that $\{q, q_i\} \subseteq \mathcal{E}_i^H(s_0)$. Since $\forall i \in \mathcal{P} : q_i \notin \Im(H, G)$, we know that $|\mathcal{I}_D(s_0)| \leq n - |\mathcal{P}| < n - k + 1$.

Next, we show by contradiction that $\forall t \in \overline{\{s_0\}} \cap \mathcal{L}(H) : |\mathcal{I}_D(t)| < n - k + 1$. Let us assume that $\exists t \in \overline{\{s_0\}} \cap \mathcal{L}(H) : |\mathcal{I}_D(t)| \geq n - k + 1$, i.e., there exists a set $\mathcal{Q} \subseteq \mathcal{I}$ such that $\forall i \in \mathcal{Q} : \mathcal{E}_i^H(t) \subseteq \Im(H, G)$ and $|\mathcal{Q}| \geq n - k + 1$. By the definition of $\Im(H, G)$, we know that any state reached from an indicator state is an indicator state. Therefore, we know that $\forall i \in \mathcal{Q} : \mathcal{E}_i^H(s_0) \subseteq \Im(H, G)$, which implies that $|\mathcal{I}_D(s_0)| \geq |\mathcal{Q}| \geq n - k + 1$. This is a contradiction. Hence, we know that $\forall t \in \overline{\{s_0\}} \cap \mathcal{L}(H) : |\mathcal{I}_D(t)| < n - k + 1$.

Finally, since q is a boundary state, we know that there exists an event $\sigma \in \Sigma$ such that string $s_0\sigma \in \mathcal{L}(G) \setminus \mathcal{L}(H)$ violates the specification. Note that $\overline{\{s_0\sigma\}} \cap \mathcal{L}(H) = \overline{\{s_0\}} \cap \mathcal{L}(H)$. Therefore, we have that

$$(\exists s_0\sigma \in \mathcal{L}(G) \setminus \mathcal{L}(H))\left(\forall t \in \overline{\{s_0\sigma\}} \cap \mathcal{L}(H)\right)[|\mathcal{I}_D(t)| < n - k + 1]$$

i.e., $\mathcal{L}(H)$ is not k-reliably coprognosable.

($\Leftarrow$) By contrapositive. Suppose that $\mathcal{L}(H)$ is not k-reliably coprognosable. We know that there exists a string $s \in \mathcal{L}(G) \setminus \mathcal{L}(H)$ such that $\forall t \in \overline{\{s\}} \cap \mathcal{L}(H) : |\mathcal{I}_D(t)| < n - k + 1$. Let us consider the longest

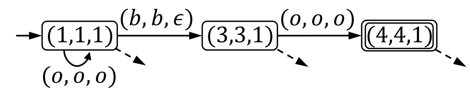

Fig. 3. Part of the k-verifier for H and G in Fig. 2 with $k = 1$, $\Sigma_{o,1} = \{a, b, o\}$, and $\Sigma_{o,2} = \{a, o\}$.

t in $\overline{\{s\}} \cap \mathcal{L}(H)$, i.e., $\delta_H(t) \in \partial(H, G)$. Since $|\mathcal{I}_D(t)| < n - k + 1$, we know that there exists a set of agents $\mathcal{P} \subseteq \mathcal{I}$ such that:
1) $\forall i \in \mathcal{P} : \mathcal{E}_i^H(t) \not\subseteq \Im(H, G)$;
2) $|\mathcal{P}| \geq k$.

By condition 1), we know that

$$(\forall i \in \mathcal{P})(\exists t_i \in \mathcal{L}(H))[P_i(t) = P_i(t_i) \wedge \delta_H(t_i) \in \Upsilon(H, G)].$$

Let us consider string $w = (t, w_1, \dots, w_n)$, where $w_i = t_i$ if $i \in \mathcal{P}$ and $w_i = t$ if $i \in \mathcal{I} \setminus \mathcal{P}$. Such a string is defined in V, since $\forall i \in \mathcal{I} : P_i(t) = P_i(w_i)$, and we denote by $x_w = (q, w_1, \dots, w_n) = f_V(x_{0,V}, w)$ the state reached upon w in V. Since $\forall i \in \mathcal{P} : \delta_H(t_i) \in \Upsilon(H, G)$, by condition 2), we know that

$$|\{i \in \mathcal{I} : q_i \in \Im(H, G)\}| \leq n - |\mathcal{P}| < n - k + 1.$$

Moreover, since $q = \delta_H(t) \in \partial(H, G)$, we know that $x_w \in X_{m,V}$, i.e., $\mathcal{L}_m(V) \neq \emptyset$. ■

Example 2: Let us return to Example 1 and consider automata G and H shown in Fig. 2. First, we still suppose that there are two local prognosers whose observations are $\Sigma_{o,1} = \{a, b, o\}$, and $\Sigma_{o,2} = \{a, o\}$, respectively. Let $k = 1$. Then part of the corresponding k-verifier constructed from H and G is shown in Fig. 3. For example, at the initial state $(1, 1, 1)$, event (b, b, ϵ) is defined according to (14) and state $(3, 3, 1)$ is reached via this event. From state $(3, 3, 1)$, state $(4, 4, 1)$ is reached via event (o, o, o). However, state $\{4, 4, 1\}$ is a marked state, since $4 \in \partial(H, G)$ and $|\{4, 1\} \cap \Im(H, G)| = |\{4\}| = 1 < 2 - 1 + 1 = 2$. Therefore, we know that $\mathcal{L}(H)$ is not one-reliably coprognosable with respect to $\mathcal{L}(G)$, $\Sigma_{o,1}$, and $\Sigma_{o,2}$.

One way to resolve this unreliability issue is to add more local prognosers in order to increase the reliability of the entire system. Suppose that there is still only one unreliable local prognostic decision at each step, but we add a new local prognoser with observable events set $\Sigma_{o,3} = \{b, o\}$. Therefore, we are interested in verifying whether or not the new system with three local prognosers is two-reliably coprognosable. In order to do so, we construct the complete k-verifier for $G, H, \Sigma_{o,i}, i \in \{1, 2, 3\}$, and $k = 2$, which is shown in Fig. 4. We see that there is no marked state in V, since for each state whose first component is 4, i.e., the unique boundary state, there are at least two state components that are indicator states. Therefore, we know that $\mathcal{L}(H)$ is two-reliably coprognosable with respect to $\mathcal{L}(G)$, $\Sigma_{o,1}$, $\Sigma_{o,2}$, and $\Sigma_{o,3}$.

Remark 3: In the k-verifier V, there are at most $|Q|^{n+1}$ states and $(1 + n)|\Sigma||Q|^{n+1}$ transitions. Moreover, the complexity of checking whether or not $\mathcal{L}_m(V)$ is linear in the number of states and the number of transitions in V. Therefore, we know that the total worst-case complexity of verifying k-reliable coprognosability is $O(n|\Sigma||Q|^{n+1})$, which is the same as the complexity of the algorithm in [12] for checking coprognosability. In other words, by using the k-verifier proposed, we can verify k-reliability in addition to the verification of coprognosability without spending additional cost.

V. Computation of Reactive Bound

So far, we have studied the notion of k-reliable coprognosability and its verification. Note that, k-reliable coprognosability only guarantees that faults can be predicted prior to their occurrences.

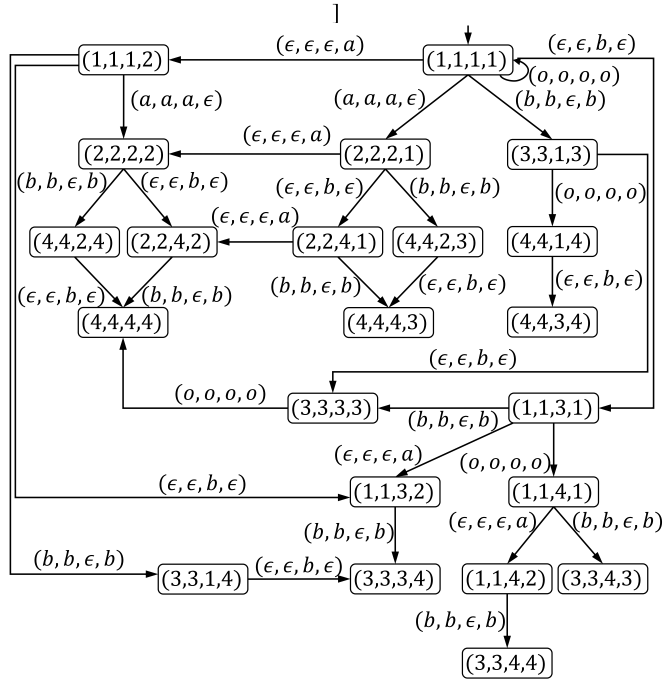

Fig. 4. Complete k-verifier for H and G in Fig. 2 with $k = 2$, $\Sigma_{o,1} = \{a, b, o\}$, $\Sigma_{o,2} = \{a, o\}$, and $\Sigma_{o,3} = \{b, o\}$.

However, in many applications, this condition may not be enough, since we need to predict faults "as early as possible" such that some actions can be taken in order to protect the system or to prevent faults from happening. Therefore, if the system is k-reliably coprognosable, one may also be interested in knowing how early the prognoser can react to the occurrences of faults. In order to address this issue, we introduce the notion of k-reliable reactive bound.

For any string $s \in \mathcal{L}(H)$ such that $\delta_H(s) \in \partial(H, G)$, we denote by $\rho(s)$ the number of steps between the time when the first fault alarm is reliably issued and the time when the first fault occurs, that is

$$\rho(s) = \max_{t\in\overline{\{s\}}\ :\ |\mathcal{I}_D(t)|\geq n-k+1} |s \setminus t| \tag{17}$$

where $s \setminus t$ denotes the string in s after t, i.e., $t(s \setminus t) = s$. We call $\rho(s)$ the k-reliable reactive bound of string s. We note that, if $|\mathcal{I}_D(t)| \geq n - k + 1$, then for any of its extensions $tv \in \overline{\{s\}}, v \in \Sigma^*$, we have that $|\mathcal{I}_D(tv)| \geq n-k+1$. Then, the k-reliable reactive bound for H and G, denoted by $\rho(H, G)$, is defined as the worst (minimal) k-reliable reactive bound of strings in $\mathcal{L}(H)$, that is

$$\rho(H, G) := \min_{s\in\mathcal{L}(H)\ :\ \delta_H(s)\in\partial(H,G)} \rho(s). \tag{18}$$

Next, we show that how to compute the k-reliable reactive bound. First, we introduce some necessary notions.

Let $q \in Q_H$ be a state in H. We say that q is an unprognosable state if there exists a state $(q, q_1, \ldots, q_n) \in X_V$ in V, such that $|\{i \in \mathcal{I} : q_i \in \Upsilon(H, G)\}| \geq k$. We denote by $\Omega(H, G) \subseteq Q_H$ the set of unprognosable states in H. We say that q is a prognosable state if it is not an unprognosable state. The notion of unprognosable state is a generalization of the notion of unprognosable state defined in [12]. Clearly, we know that, if $q \in \Omega(H, G)$, then there exists a string $s \in \mathcal{L}(H) : \delta_H(s) = q$ such that $|\mathcal{I}_D(s)| < n - k + 1$. Then, for any unprognosable state $q \in \Omega(H, G)$, we denote by $\eta(q)$, the set of strings from q to a boundary state without visiting any unprognosable state expect q. Formally, we have that

$$\eta(q) = \big\{s \in \mathcal{L}(H, q)\ :\delta_H(q, s) \in \partial(H, G) \wedge \forall t \in \overline{\{s\}} \text{ s.t. } \delta_H(q, t) \notin \Omega(H, G)\big\}. \tag{19}$$

Note that $\eta(q)$ may be empty for some $q \in \Omega(H, G)$. Now, we are ready to present the key theorem that reveals how the k-reliable reactive bound can be computed.

Theorem 3: Let H be the specification automaton and G be the system automaton. We have that

$$\rho(H, G) = \min_{q\in\Omega(H,G), s\in\eta(q)} \{|s| - 1\}. \tag{20}$$

Proof: We prove this theorem by the following two parts.

LHS≥RHS: Suppose that (18), i.e., the LHS of (20), attains minimum by string $\hat{s} \in \mathcal{L}(H)$ and $\hat{t}$ is the corresponding string such that $|\hat{s} \setminus \hat{t}| = \max_{t\in\overline{\{\hat{s}\}}\ :\ |\mathcal{I}_D(t)|\geq n-k+1} |\hat{s} \setminus t|$.

We claim that any prefix of $\hat{s}$ containing $\hat{t}$ reaches a prognosable state, i.e., $(\forall \hat{t}v \in \overline{\{\hat{s}\}} : v \in \Sigma^*)[\delta_H(\hat{t}v) \notin \Omega(H, G)]$. To see this, we assume that $\delta_H(\hat{t}v)$ is an unprognsable state. Then, we know that

$$\big(\exists u \in \mathcal{L}(H)\ \delta_H(u) = \delta_H(\hat{t}v)\big) [|\mathcal{I}_D(u)| < n - k + 1 \wedge \delta_H(u(\hat{s} \setminus (\hat{t}v))) \in \partial(H, G).$$

This implies that the k-realiable reactive bound for string $u(\hat{s}\setminus(\hat{t}v))$ is at most $|\hat{s}\setminus\hat{t}|-1$, which is strictly smaller than $|\hat{s}\setminus\hat{t}|$. This contradicts the fact that (18) attains minimum at $\hat{s} \in \mathcal{L}(H)$. Therefore, we know that $(\forall \hat{t}v \in \overline{\{s\}} : v \in \Sigma^*)[\delta_H(\hat{t}v) \notin \Omega(H, G)]$.

Moreover, we write string $\hat{t}$ in the form of $\hat{t} = t'\sigma, \sigma \in \Sigma$, i.e., σ is the last event in $\hat{t}$. Then we know that $|\mathcal{I}_D(t')| < n-k+1$, since $\hat{t}$ is the shortest prefix of s such that $|\mathcal{I}_D(\hat{t})| \geq n - k + 1$. This implies that $\delta_H(t')$ is an unprognosable state. Since $\forall \hat{t}v \in \overline{\{\hat{s}\}}, v \in \Sigma^*, \delta_H(\hat{t}v)$ is a prognsable state, by considering $q = \delta_H(t'), s = \hat{s} \setminus t'$ for the RHS of (20), we know that

$$\min_{q\in\Omega(H,G), s\in\eta(q)} \{|s| - 1\} \leq |\hat{s} \setminus t'| - 1 = |\hat{s} \setminus \hat{t}| = \rho(H, G).$$

LHS≤RHS: Suppose that the RHS of (20) attains minimum at $\hat{q} \in \Omega(H, G), \hat{s} \in \eta(\hat{q})$. Since $\hat{q}$ is an unprognosable state, we know that there exists a string $w \in \mathcal{L}(H)$ such that $\delta_H(w) = \hat{q}$ and $|\mathcal{I}_D(w)| < n - k + 1$. We write $\hat{s}$ in the form of $\sigma s', \sigma \in \Sigma$, i.e., σ is the first event of string $\hat{s}$. Clearly, we know that $\forall v \in \Sigma^* : w\sigma v \in \overline{\{s\}} \Rightarrow |\mathcal{I}_D(w\sigma v)| \geq n - k + 1$. For otherwise, if $\exists w\sigma v \in \overline{\{s\}} : |\mathcal{I}_D(w\sigma v)| < n - k + 1$, then it contradicts the fact that $\mathcal{I}_D(w\sigma v)$ is a prognosable state. Moreover, since $\hat{s} \in \eta(\hat{q})$, we know that $\delta_H(w\hat{s}) \in \partial(H, G)$. Therefore, by considering $s = w\hat{s}$ and $t = w\sigma$ for the LHS of (20), i.e., (18), we know that

$$\rho(H, G) \leq |w\hat{s} \setminus w\sigma| = |\hat{s}| - 1 = \min_{q\in\Omega(H,G), s\in\eta(q)} \{|s| - 1\}.$$

Since LHS≤RHS and RHS≤LHS, we know that (20) holds. ■

Remark 4: The above theorem says that, in order to compute the k-reliable reactive bound, it suffices to find the shortest path from an unprognosable state to a boundary state via some prognosable states. Note that determining set $\Omega(H, G)$ can be done in $O(|\Sigma||Q|^{n+1})$ by searching the state space of the k-verifier V. The complexity of the shortest path search problem is linear in the size of H. Therefore, the k-reliable reactive bound $\rho(H, G)$ can be computed in $O(|\Sigma||Q|^{n+1})$.

The following example illustrates how the k-reliable reactive bound $\rho(H, G)$ is computed.

Example 3: Let us revisit automata G and H shown in Fig. 2 and assume that there are three local prognosers whose observations are $\Sigma_{o,1} = \{a, b, o\}, \Sigma_{o,2} = \{a, o\}$, and $\Sigma_{o,3} = \{b, o\}$, respectively. Let $k = 2$. We have shown in Example 2 that $\mathcal{L}(H)$ is two-reliably coprognosable with respect to G and $\Sigma_{o,i}, i \in \{1, 2, 3\}$ and the complete k-verifier has been shown in Fig. 4. By searching states in V, we know that $\Omega(H, G) = \{1\}$, since for state $q = (1, q_1, q_2, q_3) = (1, 1, 1, 2)$, we have that $|\{i \in \mathcal{I} : q_i \in \Upsilon(H, G)\}| = 2 \geq k$. And we know that states $2 - 4$ in H are all prognosable state. Therefore, the shortest path from the unique unprognosable state 1 to the unique boundary state 4 via prognosable states is *ab* or *bo* whose lengths are both 2. Therefore, we know that $\rho(H, G) = 2 - 1 = 1$, i.e., all faults are guaranteed to be reliably predicted one step before their occurrences.

VI. Conclusion

In this correspondence paper, we studied the problem of reliable decentralized fault prognosis of discrete-event systems. The notion of k-reliable coprognosability was introduced as the necessary and sufficient condition for the existence of a decentralized prognoser under the presence of unreliable local prognostic decisions. Both the verification of k-reliable coprognosability and the computation of k-reliable reactive bound were investigated. These results extend previous studies on decentralized fault prognosis, in which all local prognostic decisions are assumed to be reliable.

In this paper, reliability was defined in the sense of decision losses. However, in some applications, it is possible that a local prognostic decision may arrive at the coordinator after some finite delay. Therefore, investigating the effect of decision delays is an important issue in the future. Also, in this paper, we adopted the disjunctive architecture for the decentralized prognosis problem. Investigating the reliability issue under other decentralized architectures, e.g., conjunctive architecture [11] or inference-based architecture [24], is also an interesting future direction.

References

[1] J. C. Basilio and S. Lafortune, "Robust codiagnosability of discrete event systems," in *Proc. Amer. Control Conf.*, St. Louis, MO, USA, 2009, pp. 2202–2209.

[2] L. K. Carvalho, J. C. Basilio, and M. V. Moreira, "Robust diagnosis of discrete event systems against intermittent loss of observations," *Automatica*, vol. 48, no. 9, pp. 2068–2078, 2012.

[3] C. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, 2nd ed. New York, NY, USA: Springer, 2008.

[4] F. Cassez and A. Grastien, "Predictability of event occurrences in timed systems," in *Formal Modeling and Analysis of Timed Systems*. Berlin, Germany: Springer, 2013, pp. 62–76.

[5] M. Chang, W. Dong, Y. Ji, and L. Tong, "On fault predictability in stochastic discrete event systems," *Asian J. Control*, vol. 15, no. 5, pp. 1458–1467, 2013.

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

[7] H. Cho and S. I. Marcus, "On supremal languages of classes of sublanguages that arise in supervisor synthesis problems with partial observation," *Math. Control Signal Syst.*, vol. 2, no. 1, pp. 47–69, 1989.

[8] S. Genc and S. Lafortune, "Predictability of event occurrences in partially-observed discrete-event systems," *Automatica*, vol. 45, no. 2, pp. 301–311, 2009.

[9] T. Jéron, H. Marchand, S. Genc, and S. Lafortune, "Predictability of sequence patterns in discrete event systems," in *Proc. 17th IFAC World Congr.*, Seoul, Korea, 2008, pp. 537–543.

[10] A. Khoumsi and H. Chakib, "Multi-decision decentralized prognosis of failures in discrete event systems," in *Proc. Amer. Control Conf.*, St. Louis, MO, USA, 2009, pp. 4974–4981.

[11] A. Khoumsi and H. Chakib, "Conjunctive and disjunctive architectures for decentralized prognosis of failures in discrete-event systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 2, pp. 412–417, Apr. 2012.

[12] R. Kumar and S. Takai, "Decentralized prognosis of failures in discrete event systems," *IEEE Trans. Autom. Control*, vol. 55, no. 1, pp. 48–59, Jan. 2010.

[13] D. Lefebvre, "Fault diagnosis and prognosis with partially observed Petri nets," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 10, pp. 1413–1424, Oct. 2014.

[14] F. Lin, "Control of networked discrete event systems: Dealing with communication delays and losses," *SIAM J. Control Optim.*, vol. 52, no. 2, pp. 1276–1298, 2014.

[15] F. Liu and Z. Dziong, "Reliable decentralized control of fuzzy discrete-event systems and a test algorithm," *IEEE Trans. Cybern.*, vol. 43, no. 1, pp. 321–331, Feb. 2013.

[16] F. Liu and H. Lin, "Reliable supervisory control for general architecture of decentralized discrete event systems," *Automatica*, vol. 46, no. 9, pp. 1510–1516, 2010.

[17] S. Nakata and S. Takai, "Reliable decentralized failure diagnosis of discrete event systems," *SICE J. Control Meas. Syst. Integr.*, vol. 6, no. 5, pp. 353–359, 2013.

[18] F. Nouioua, P. Dague, and L. Ye, "Probabilistic analysis of predictability in discrete event systems," in *Proc. 25th Int. Workshop Principle Diagn.*, Graz, Austria, 2014.

[19] S.-J. Park, "Robust and nonblocking supervisory control of nondeterministic discrete event systems with communication delay and partial observation," *Int. J. Control*, vol. 85, no. 1, pp. 58–68, 2012.

[20] S.-J. Park and K.-H. Cho, "Supervisory control of discrete event systems with communication delays and partial observations," *Syst. Control Lett.*, vol. 56, no. 2, pp. 106–112, 2007.

[21] A. Saboori and S. H. Zad, "Robust nonblocking supervisory control of discrete-event systems under partial observation," *Syst. Control Lett.*, vol. 55, no. 10, pp. 839–848, 2006.

[22] S. Shu and F. Lin, "Decentralized control of networked discrete event systems with communication delays," *Automatica*, vol. 50, no. 8, pp. 2108–2112, 2014.

[23] S. Takai, "Robust prognosability for a set of partially observed discrete event systems," *Automatica*, vol. 51, pp. 123–130, Jan. 2015.

[24] S. Takai and R. Kumar, "Inference-based decentralized prognosis in discrete event systems," *IEEE Trans. Autom. Control*, vol. 56, no. 1, pp. 165–171, Jan. 2011.

[25] S. Takai and R. Kumar, "Distributed failure prognosis of discrete event systems with bounded-delay communications," *IEEE Trans. Autom. Control*, vol. 57, no. 5, pp. 1259–1265, May 2012.

[26] S. Takai and T. Ushio, "Reliable decentralized supervisory control of discrete event systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 5, pp. 661–667, Oct. 2000.

[27] S. Takai and T. Ushio, "Reliable decentralized supervisory control for marked language specifications," *Asian J. Control*, vol. 5, no. 1, pp. 160–167, 2003.

[28] T. Yamamoto and S. Takai, "Reliable decentralized diagnosis of discrete event systems using the conjunctive architecture," *IEICE Trans. Fund. Electron. Commun. Comput. Sci.*, vol. 97, no. 7, pp. 1605–1614, 2014.

[29] L. Ye, P. Dague, and F. Nouioua, "Predictability analysis of distributed discrete event systems," in *Proc. 52nd Conf. Decis. Control*, Firenze, Italy, 2013, pp. 5009–5015.