Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 58-1 (2024) 174–179

Better Late than Never: On Epistemic Diagnosability of Discrete Event Systems [⋆]

Bohan Cui [*] **Ziyue Ma** [**] **Alessandro Giua** [***] **Xiang Yin** [*]

[*] *Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China. (E-mail: {bohan_cui, yinxiang}@sjtu.edu.cn).*
[**] *School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China. (E-mail: maziyue@xidian.edu.cn).*
[***] *DIEE, University of Cagliari, Cagliari 09123, Italy. (E-mail: giua@unica.it).*

Abstract: We investigate the diagnosability verification problem in the framework of discrete-event systems. Most of the existing works on this topic assume that faults are related to the internal behaviors of the system such as occurrences of particular events. In this work, motivated by information-flow security considerations, we model faults as some critical information leakages of the system to an intruder, which may have different observations from the system user. Specifically, we say that a fault occurs if the intruder knows that the system has passed by a secret state. We present a formal notion called *epistemic diagnosability* to capture whether or not the system user can always detect, based on its own observation, the critical information leakage to an intruder within a bounded delay. We show that this new notion subsumes the standard notion of event-based diagnosability. Furthermore, an effective algorithm is provided to verify this new notion.

Keywords: Discrete Event Systems, Diagnosis, Security, Partial Observation.

1. INTRODUCTION

With the advancement of network and information technologies, cyber-physical systems (CPSs) have become extensively used in our society. While CPSs enhance the flexibility and intelligence of engineering systems, they concurrently introduces increased complexity. Large-scale CPSs, in particular, face the dual challenges of vulnerability to failures and potential leakage of critical information due to their intricate operation logic and frequent data interactions. Therefore, to ensure the performance of safety-critical CPSs, fault diagnosis and security analysis have become more crucial and have drawn a lot of attention in recent years (Basilio et al., 2021; Liu et al., 2022).

In this paper, we consider the fault diagnosis problem in the context of discrete-event systems (DESs), a class of dynamic systems with discrete-state spaces and event-triggered dynamics widely used in describing high-level behaviors of CPSs (Cassandras and Lafortune, 2008). In the context of fault diagnosis, it is assumed that the system may experience *faults* during its operation. Then a system is said to be *diagnosable* if the system user can always detect the occurrences of faults based on its own observations within a finite delay (Sampath et al., 1995). During the past few years, fault diagnosis of DES has drawn a lot of interest due to its importance; see, e.g., Yin et al. (2019); Carvalho et al. (2021); Takai (2021); Ma et al. (2023); Dong et al. (2023b); Zhao et al. (2024).

Since diagnosability requires the system user to always be able to detect the occurrence of fault, one key ingredient in this notion is what is a fault and how to describe its occurrence. In the context of diagnosability analysis, one of the most commonly used approaches is to model faults as the occurrences of particular events or transitions (Lefebvre and Delherm, 2007; Basile et al., 2014). An equivalent approach is to consider state-based fault, where the system is faulty if it has visited a fault state (Zad et al., 2003). In a more general setting, faults can be defined as the executions of incorrect sequences of events, i.e., behavior patterns, (Jéron et al., 2006; Yin and Lafortune, 2017; Pencolé and Subias, 2021; Ma et al., 2023). There are also works in the literature describing faults as the violations of logical formulae such as linear temporal logic or metric interval temporal logic; see, e.g., Jiang and Kumar (2004); Chen and Kumar (2015); Dong et al. (2023a).

Note that the previously mentioned concepts of faults are directly associated with the internal behavior of systems. However, in certain applications, faults can also be linked to the *external knowledge* of the system based on its information-flow. To illustrate this, consider the information-flow security property of DES known as opacity. This property demands that the system maintains plausible deniability against a passive intruder with respect to some critical information, such as secret states. A violation of opacity implies the existence of a trajectory generated by the system where critical information is

[⋆] This work was supported by the National Natural Science Foundation of China (62173226,6206113600). This work was partially supported by project SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union - NextGenerationEU.

10.1016/j.ifacol.2024.07.030

disclosed to the intruder. Therefore, from a security point of view, such leakage of critical information can also be viewed as a fault. Detecting such information leakage in a timely manner becomes essential to prevent potential catastrophic consequences.

In this paper, we revisit the diagnosability verification problem from a new perspective. Specifically, we assume that the system is under observation from two distinct entities: the system user and the intruder, each having its own information-flow. The intruder aims to expose critical information of the system, while the user seeks to identify such information leakage as they occur. Drawing inspiration from the concept of strong infinite-step opacity (Falcone and Marchand, 2015; Ma et al., 2021), we represent critical information leakage as detecting if the system has visited specific secret states, which is considered as a fault in our setting. We formalize this awareness of information leakages as the notion of "*epistemic diagnosability*". Formally, a system is considered K- epistemically diagnosable if the user can detect, in no more than K steps, that the intruder has detected the system's secret. We provide an effective algorithm for verifying this new notion of diagnosability.

Our work is conceptually related to the field of epistemic logic within knowledge theory (Van Ditmarsch et al., 2007), where properties based on knowledge relationships, such as "you know that I know", can be formalized. However, in epistemic logic, these properties are typically interpreted by using a Kripke model. To the best of our knowledge, diagnosis of faults based on epistemic considerations has not been explored in the context of DES. A related concept, termed high-order opacity, has been investigated in our previous work (Cui et al., 2022). Nevertheless, this notion primarily addresses the requirement to secure the knowledge of the system user, whereas our present paper focuses on the awareness of information leakages. These two notions are not equivalent to each other.

The rest of this paper is organized as follows. Section 2 provides some basic preliminaries. Then, we introduce the notion of *epistemic diagnosability* in Section 3. The verification procedure for this notion is outlined in Section 4. Finally, we conclude the paper and discuss future directions in Section 5.

2. PRELIMINARIES

2.1 System Model

Let Σ be a finite set of events. A string is a finite sequence of events and Σ^* denotes the set of all strings over Σ including the empty string ϵ. For any string $s \in \Sigma^*$, $|s|$ denotes the length of s with $|\epsilon| = 0$. A language $L \subseteq \Sigma^*$ is a set of strings. For any string $s \in L$, we denote by L/s the post-language of s in L, i.e., $L/s := \{w \in \Sigma^* : sw \in L\}$. Also, we denote by $\overline{L}$ the prefix-closure of language L, i.e., $\overline{L} = \{s \in \Sigma^* : \exists w \in \Sigma^* \text{ s.t. } sw \in L\}$.

We consider a DES modeled by a deterministic finite-state automaton (DFA)

$$G = (X, \Sigma, \delta, x_0),$$

where X is a finite set of states, Σ is a finite set of events, $\delta : X \times \Sigma \to X$ is the partial transition function, where for any $x, x' \in X$, $\sigma \in \Sigma$, $x' = \delta(x, \sigma)$ means that there exists a transition from state x to state x' via event σ, and $x_0 \in X$ is the initial state. The transition function is also extended to $\delta : X \times \Sigma^* \to X$ recursively by: (i) for any $x \in X$, $\delta(x, \epsilon) = x$ and (ii) for any $x \in X, s \in \Sigma^*, \sigma \in \Sigma$, we have $\delta(x, s\sigma) = \delta(\delta(x, s), \sigma)$. The set of all strings generated by G starting from state $x \in X$ is defined as $\mathcal{L}(G, x) = \{s \in \Sigma^* : \delta(x, s)!\}$, where "!" means "is defined". The set of all strings generated by G is defined as $\mathcal{L}(G) := \mathcal{L}(G, x_0)$. For any $s \in \mathcal{L}(G)$, we write $\delta(x_0, s)$ simply as $\delta(s)$. For the sake of simplicity, we assume that system G is live, i.e., for any $x \in X$, there exists $\sigma \in \Sigma$ such that $\delta(x, \sigma)!$ holds.

In a partially observed system, the occurrence of each event is imperfectly observed through an observation function defined as follows

$$\mathcal{H} : \Sigma \to \Delta \dot{\cup} \{\epsilon\},$$

where Δ is a new set of observation symbols. That is, we observe $\mathcal{H}(\sigma)$ upon the occurrence of event $\sigma \in \Sigma$. We say event $\sigma \in \Sigma$ is observable if $\mathcal{H}(\sigma) \in \Delta$ and unobservable if $\mathcal{H}(\sigma) = \epsilon$. The observation function is also extended to $\mathcal{H} : \Sigma^* \to \Delta^*$ as follows: for any $s \in \Sigma^*$, $\mathcal{H}(s)$ is obtained by replacing each event σ in string s with $\mathcal{H}(\sigma)$.

2.2 Event-Based Fault Diagnosis

In the context of fault diagnosis of DES, it is assumed that the system is subject to faults modeled by a set of fault events $\Sigma_f \subseteq \Sigma$. We say a string $s \in \mathcal{L}(G)$ is faulty if it contains at least one fault event, denoted as $\Sigma_f \cap s \neq \emptyset$. We define $\Psi(\Sigma_f)$ as the set of all finite faulty strings where fault events occur for the first time, i.e.,

$$\Psi(\Sigma_f) := \{s \in \mathcal{L}(G) : [\Sigma_f \cap s \neq \emptyset] \wedge [\forall t \in \overline{\{s\}} \setminus \{s\} : \Sigma_f \cap t = \emptyset]\}.$$

To capture whether or not fault events can always be detected in a finite number of steps after their occurrences, the notion of diagnosability has been proposed. Formally, the definition of K-diagnosability is as follows.

Definition 1. (Diagnosability). A system G is said to be K-diagnosable w.r.t. function $\mathcal{H}$ and fault events Σ_f if

$$\begin{aligned} &(\forall s \in \Psi(\Sigma_f))(\forall t \in \mathcal{L}(G)/s : |t| \geq K) \\ &(\forall w \in \mathcal{L}(G))[\mathcal{H}(st) = \mathcal{H}(w) \Rightarrow \Sigma_f \cap w \neq \emptyset]. \end{aligned} \quad (1)$$

3. NOTION OF EPISTEMIC DIAGNOSABILITY

In this section, we first formalize the information leakage, which is considered as fault in this paper. Then we provide the notion of epistemic diagnosability.

The issue of epistemic diagnosability arises when there are two different observation sites of the underlying system:

- the *user*, who observes the system through function $\mathcal{H}_o : \Sigma \to \Delta_o \dot{\cup} \{\epsilon\}$; and
- the *intruder*, who observes the system through function $\mathcal{H}_a : \Sigma \to \Delta_a \dot{\cup} \{\epsilon\}$.

We denote the sets of observable events of the user and the intruder as Σ_o and Σ_a, respectively, and their unobservable events are denoted by $\Sigma_{uo} := \Sigma \setminus \Sigma_o$ and $\Sigma_{ua} := \Sigma \setminus \Sigma_a$, respectively. Here we do not assume any relationship between Σ_a and Σ_o, i.e., they can be incomparable in general.

3.1 Critical Information Leakage

During the operation of the system, the system can release some information to the intruder based on its observation function $\mathcal{H}_a$. However, such information release might be critical to the system user and some critical information should not be revealed to the intruder. In general, the leakage of critical information can be defined as a predicate on the system languages

$$\texttt{rev}_a : \mathcal{L}(G) \to \{\texttt{true}, \texttt{false}\}$$

such that "$\texttt{rev}_a(s) = \texttt{true}$" indicates that the intruder knows some secret based on the observation $\mathcal{H}_a(s)$, where $s \in \mathcal{L}(G)$ is the string actual occurred. Therefore, for any two strings $s, t \in \mathcal{L}(G)$ such that $\mathcal{H}_a(s) = \mathcal{H}_a(t)$, we always have $\texttt{rev}_a(s) = \texttt{rev}_a(t)$.

In this paper, we describe the critical information leakage motivated by the notion of *strong infinite-step opacity* (Falcone and Marchand, 2015; Ma et al., 2021). Specifically, we assume that the critical information of the system user is captured by a set of secret states $X_S \subseteq X$. The leakage of the critical information is defined as follows.

Definition 2. (Critical Information Leakage). Given G, a set of secret states X_S and an intruder's observation function $\mathcal{H}_a$, for any string $s \in \mathcal{L}(G)$ generated by the system, we have $\texttt{rev}_a(s) = \texttt{true}$ iff

$$(\forall r \in \mathcal{L}(G) : \mathcal{H}_a(r) = \mathcal{H}_a(s))(\exists w \in \overline{\{r\}})[\delta(w) \in X_S]. \quad (2)$$

Intuitively, critical information leakage captures the situation where the intruder knows for sure that the system has visited a secret state based on its own observation.

Remark 1. We note that the critical information leakage considered here is irreversible. That is, for any $s \in \mathcal{L}(G)$, if $\texttt{rev}_a(s) = \texttt{true}$, then we have $\texttt{rev}_a(st) = \texttt{true}$ for all $t \in \mathcal{L}(G)/s$. This is because that, by using further observations, the intruder can only exclude impossible strings from the previous estimate. Therefore, by observing $\mathcal{H}_a(s)$, if all possible strings of the system have been passing through secret states, then it will be the same case by further observing $\mathcal{H}_a(st)$.

3.2 Epistemic Diagnosability

Before formally introducing the definition of epistemic diagnosability, we first summarize the capabilities of the system user:

- It knows the DFA model G of the system;
- It can observe the occurrence of each event in Σ_o generated online through function $\mathcal{H}_o$;
- It knows that the intruder can observe the occurrence of each event in Σ_a online through $\mathcal{H}_a$, but it cannot observe the occurrences of events $\Sigma_a \setminus \Sigma_o$.

The notion of epistemic diagnosability follows the basic idea of diagnosability, which is to detect the occurrence of "fault events" within a finite horizon. Differently, in the setting of epistemic diagnosability, "fault events" are captured by epistemic behaviors of the system, i.e., the leakage of critical information. We define $\Psi(\texttt{rev}_a)$ as the set of all finite strings where the critical information leakage occurs for the first time, i.e.,

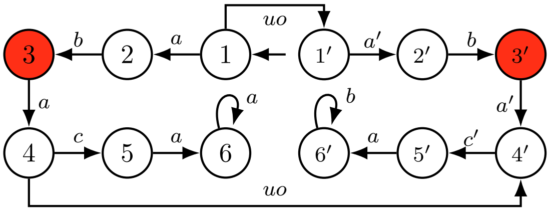

Fig. 1. System G with $X_S = \{3, 3'\}$.

$$\Psi(\texttt{rev}_a) := \left\{ se_r \in \mathcal{L}(G) : \begin{array}{c} e_r \in \Sigma \wedge \texttt{rev}_a(se_r) = \texttt{true} \wedge \\ \texttt{rev}_a(s) = \texttt{false} \end{array} \right\}$$

Then the definition of epistemic diagnosability is given as follows.

Definition 3. (Epistemic Diagnosability). We say system G is *epistemically K-diagnosable* w.r.t. $\mathcal{H}_o$, $\mathcal{H}_a$ and secret states $X_S \subseteq X$ if

$$\begin{aligned} &(\forall s \in \Psi(\texttt{rev}_a))(\forall t \in \mathcal{L}(G)/s : |t| \geq K) \\ &(\forall w \in \mathcal{L}(G))[\mathcal{H}_o(st) = \mathcal{H}_o(w) \Rightarrow \texttt{rev}_a(w) = \texttt{true}]. \quad (3) \end{aligned}$$

Epistemic diagnosability can be understood as follows. Let $s \in \mathcal{L}(G)$ be an actual string generated by the system. Upon the occurrence of s, the intruder observes $\mathcal{H}_a(s)$ and the information leakage is determined by $\mathcal{H}_a(s)$ completely. However, the user may not be immediately aware of the information leakage, and it may require further observations to detect such leakage. Suppose $t \in \mathcal{L}(G)/s$ is an actual string generated by the system thereafter, where the length of t satisfies $|t| \geq K$. Then upon the occurrence of st, from the user's point of view, the intruder may have observed any string in $\mathcal{H}_a(\mathcal{H}_o^{-1}(\mathcal{H}_o(st)) \cap \mathcal{L}(G))$. If for all string w in $\mathcal{H}_o^{-1}(\mathcal{H}_o(st)) \cap \mathcal{L}(G)$, we have $\texttt{rev}_a(w) = \texttt{true}$, then the user knows for sure that predicate $\texttt{rev}_a$ holds $\texttt{true}$ from the intruder's point of view.

We use the following example to illustrate the notion of epistemic diagnosability.

Example 1. Let's consider system G as shown in Figure 1, where observable events of the two sites are $\Sigma_o = \{a, b, c, a', c'\}$ and $\Sigma_a = \{uo, a', c'\}$, respectively. We define $\mathcal{H}_o(x) = \mathcal{H}_o(x')$ for $x \in \{a, c\}$. Given $X_S = \{3, 3'\}$, this system is not epistemically 2-diagnosable but is 3-diagnosable. To see this, let us first consider the actual string generated by the system as $uo\, a'\, b\, a'\, c'\, a\, b$. Upon the occurrence of $uo\, a'\, b\, a'$, the intruder observes $\mathcal{H}_a(uo\, a'\, b\, a') = uo\, a\, a$ and it knows that the system is at state $4'$. By making inference based on the system model, the intruder knows that the system must have passed through secret state 3 or $3'$ (in fact it knows for sure the system was at state $3'$ one step earlier).

Now, let us consider continuation $c'\, a$ with length 2. By observing $\mathcal{H}_o(uo\, a'\, b\, a'\, c'\, a) = a\, b\, a\, c\, a$, the user can infer that the actual string may also be $a\, b\, a\, c\, a$, for which the intruder observes nothing and definitely does not know the secret. Therefore, the system is not epistemically 2-diagnosable. However, by further observing $\mathcal{H}_o(c'\, a'\, b) = c\, a\, b$, which is a 3-step continuation, the user can determine the actual string for sure and therefore, it knows for sure the leakage of the secret. Similarly, for string $a\, b\, a\, uo\, c'$, where the information leakage occurs for the first time, if we consider the continuation $a\, b$, then the user can also detect the information leakage in 2 steps. Thus, the system is epistemically 3-diagnosable.

Remark 2. It is worth to note that Definition 3 requires that for any string, where the intruder knows the secret for the first time, and for any its continuations longer than K steps, the user can detect the critical information leakage for sure by observing the continuation. However, this definition does not require any detection bound for the first time the secret state has been visited from the intruder's point of view. For example, it is also possible that the intruder detects the secret N steps after the system visited secret states, and the user can detect the information leakage for sure after the secret state is visited in $K + N$ steps. This situation does not violate the requirement in Definition 3.

4. VERIFICATION OF EPISTEMIC DIAGNOSABILITY

4.1 Augmented Systems

To verify epistemic diagnosability, we first need to capture the critical information in the system, i.e., whether or not a secret state has been encountered. To this end, we augment the state-space of the original system G to encode this information in the augmented state-space. Specifically, we define the augmented system as follows.

Definition 4. (Augmented System). Given system $G = (X, \Sigma, \delta, x_0)$, the set of secret states X_S, we define the augmented system as a four-tuple

$$\tilde{G} = (\tilde{X}, \Sigma, \tilde{\delta}, \tilde{x}_0)$$

where

- $\tilde{X} \subseteq X \times \{S, N\}$ is the set of augmented states;
- Σ is the set of events;
- $\tilde{\delta} : \tilde{X} \times \Sigma \to \tilde{X}$ is the transition function defined by: for any $\tilde{x} = (x, l) \in \tilde{X}$ and $\sigma \in \Sigma$, we have $\tilde{\delta}(\tilde{x}, \sigma)!$ iff $\delta(x, \sigma)!$. Moreover, for any $\tilde{\delta}(\tilde{x}, \sigma)!$, we have
$$\tilde{\delta}(\tilde{x}, \sigma) = \begin{cases} (\delta(x, \sigma), N) & \text{if } l = N \wedge \delta(x, \sigma) \in X_{NS} \\ (\delta(x, \sigma), S) & \text{otherwise} \end{cases}$$
- $\tilde{x}_0$ is the initial augmented state which is defined by
$$\tilde{x}_0 = \begin{cases} (x_0, N) & \text{if } x_0 \in X_{NS} \\ (x_0, S) & \text{otherwise} \end{cases}$$

By definition, the above augmented system has the following properties:

- First, the augmented system generates the same language as the original system. Specifically, we have $\mathcal{L}(\tilde{G}) = \mathcal{L}(G)$.
- Second, define the set of secret augmented states as $\tilde{X}_S = \{(x, l) \in \tilde{X} : l = S\}$. Then for any $s \in \mathcal{L}(\tilde{G})$, such that $\tilde{\delta}(s) \in \tilde{X}_S$, there exists $t \in \overline{\{s\}}$ and $\tilde{\delta}(t) = (x', l')$ such that $x' \in X_S$. In other words, each string that leads to augmented state (x, l) with $l = S$ must have passed a secret state $x \in X_S$.

By constructing the augmented system, we can describe the information leakages based on the current-state estimate of the augmented system from the intruder's point of view. Formally, define the *current-augmented-state estimate* (w.r.t. $\mathcal{H}_a$) upon the observation of α as

$$\hat{\tilde{X}}_a(\alpha) = \left\{ \tilde{\delta}(s) \in \tilde{X} : \exists s \in \mathcal{L}(\tilde{G}) \text{ s.t. } \mathcal{H}_a(s) = \alpha \right\},$$

we have the following proposition.

Proposition 1. For any $s \in \mathcal{L}(G)$, we have $\texttt{rev}_a(s) = \texttt{true}$ iff $\hat{\tilde{X}}_a(\mathcal{H}_a(s)) \subseteq \tilde{X}_S$.

The above proposition suggests that the information leakage can be captured by current-state estimate of the augmented system from the intruder's perspective. Therefore, for any string $s \in \mathcal{L}(G)$, in order to track the information leakage, it is necessary to track all the strings that the intruder cannot distinguish.

Remark 3. Notice that our notion of epistemic diagnosability subsumes the standard event-based diagnosability as in defined in Definition 1. To see the reduction, one can first set $\mathcal{H}_a(\sigma) = \sigma$ for all $\sigma \in \Sigma$. Then using the augmented system, we let

$$\tilde{\delta}(\tilde{x}, \tilde{\sigma}) = \begin{cases} (\delta(x, \sigma), N) & \text{if } l = N \wedge \sigma \notin \Sigma_f \\ (\delta(x, \sigma), S) & \text{otherwise,} \end{cases}$$

and let $\tilde{x}_0 = (x_0, N)$. Since the intruder knows the current state perfectly, for any $s\sigma \in \mathcal{L}(G)$, if $\tilde{\delta}(s) \in \tilde{X}_S$, then we have $\Sigma_f \cap s \neq \emptyset$. Therefore, $\texttt{rev}_a(st) = \texttt{true}$ is equivalent to $\Sigma_f \cap st \neq \emptyset$ and $s \in \Psi(\Sigma_f)$ by Proposition 1.

4.2 Secret Recognizer

For any internal string $s \in \mathcal{L}(G)$, in order to track all possible strings indistinguishable from the intruder's point of view, we construct the secret recognizer as follows.

Definition 5. (Secret Recognizer). Given system G and secret states X_S, the secret recognizer is defined as a four-tuple

$$T = (Q, \Sigma, f, q_0)$$

where

- $Q \subseteq X \times 2^{\tilde{X}}$ is the set of states,
- Σ is the set of events,
- $q_0 = (x_0, \{\tilde{\delta}(w) : w \in \Sigma_{ua}^*\})$ is the initial state, and
- $f : Q \times \Sigma \to Q$ is the deterministic transition function defined by: for any $q = (x, y) \in Q$ and $\sigma \in \Sigma$, if $\sigma \in \Sigma_a$, we have
$$f(q, \sigma) = \left(\delta(x, \sigma), \left\{ \tilde{\delta}(\tilde{x}, w) : \begin{array}{c} \exists \tilde{x} \in y, w \in \Sigma^*, \\ \mathcal{H}_a(w) = \mathcal{H}_a(\sigma) \end{array} \right\} \right),$$
Otherwise,
$$f(q, \sigma) = (\delta(x, \sigma), y).$$

Intuitively, the secret recognizer is constructed by synchronizing the original system G with the current-state estimator of the augmented system w.r.t. $\mathcal{H}_a$. Specifically, by construction, for any $s \in \mathcal{L}(G)$, we have $f(s) = (\delta(s), \hat{\tilde{X}}_a(\mathcal{H}_a(s)))$, where the first element is the current state upon the occurrence of s, and the second element is the current-augmented-state estimate upon observation $\mathcal{H}_a(s)$.

Structure T essentially tracks whether the secret has been revealed upon any occurrence of string $s \in \mathcal{L}(G)$. To this end, we define the information leakage states by: for any $q = (x, y) \in Q$, we say q is an information leakage state iff for any $\tilde{x} = (x, l) \in y$, we have $l = S$, i.e., $\tilde{x} \in \tilde{X}_S$. We denote the set of information leakage states as $Q_R \subseteq Q$. Then we have the following proposition.

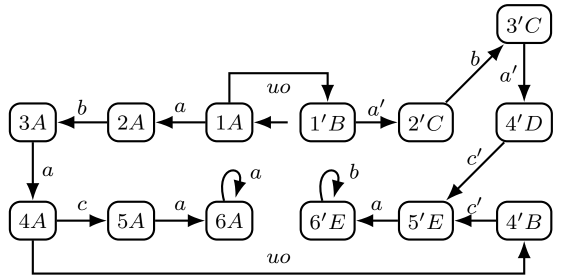

Fig. 2. Secret recognizer of the system shown in Fig. 1.

Symbol	Meaning
A	$\left\{\begin{array}{c}1N,2N,3S\\4S,5S,6S\end{array}\right\}$
B	$\{1'N,4'S\}$
C	$\{2'N,3'S\}$
D	$\{4'S\}$
E	$\{5'S,6'S\}$

Table 1. State estimates in Fig. 2.

Proposition 2. For any $s \in \mathcal{L}(G)$, we have $\texttt{rev}_a(s) = \texttt{true}$ iff $f(s) \in Q_R$.

The above proposition says that, to check whether the information is released or not, we can construct the secret recognizer and check if the corresponding state reached by $f(s)$ is a information leakage state. We illustrate the above procedure by the following example.

Example 2. Let us still consider system G as shown in Figure 1. According to the definition of augmented system and the structure of G, it is clear that the secret augmented states should be $\tilde{X}_S = \{3S, 4S, 5S, 6S, 3'S, 4'S, 5'S, 6'S\}$. We build the secret recognizer as shown in Figure 2, in which the estimate states, from A to E, are shown in Table 1. For example, to construct the initial state q_0, we have $x_0 = 1$, and $A = \{1N, 2N, 3S, 4S, 5S, 6S\}$ since $\Sigma_{ua} = \{a, b, c\}$. Then, from the initial state, consider the event uo, we have $\delta(1, uo) = 1'$. For $1N, 2N, 3S, 4S \in A$, consider strings $uo, a\,b\,a\,uo, b\,a\,uo, a\,uo$, whose projections (w.r.t. $\mathcal{H}_a$) are all uo. The augmented system $\tilde{G}$ can reach $1'N$ and $4'S$. This is why we have $f(1A, uo) = 1'B$. Note that the information leakage states are $Q_R = \{4'D, 5'E, 6'E\}$ according to the definition. For instance, consider string $s = uo\,a'\,b\,a'$, we have $f(s) \in Q_R$, i.e., $\texttt{rev}_a(s) = \texttt{true}$, which is consistent with our analysis in Example 1.

4.3 Verification Structure

Now, to verify epistemic diagnosability, we can further reason the secret recognizer from the user's perspective, i.e., to check whether it can diagnose the critical information leakage based on its own observations. Therefore, we need to further build the verification structure as follows.

Definition 6. (Verification Structure). Given system G and secret states X_S, the verification structure is a four-tuple

$$V = (Q_V, \Sigma_V, f_V, q_{0,V})$$

where

- $Q_V \subseteq Q \times Q$ is the set of states;
- $\Sigma_V = \Sigma_V^o \dot{\cup} \Sigma_V^{uo}$ is the set of events, where
  - $\Sigma_V^o = \{(\sigma_1, \sigma_2) \in \Sigma_o \times \Sigma_o : \mathcal{H}_o(\sigma_1) = \mathcal{H}_o(\sigma_2)\}$;
  - $\Sigma_V^{uo} = \{(\sigma_1, \epsilon) : \sigma_1 \in \Sigma_{uo}\} \cup \{(\epsilon, \sigma_2) : \sigma_2 \in \Sigma_{uo}\}$;
- $f_V : Q_V \times \Sigma_V \to Q_V$ is the transition function defined by: for any $q_V = (q_1, q_2) \in Q_V$ and $\sigma_V = (\sigma_1, \sigma_2) \in \Sigma_V$, we have
$$f_V(q_V, \sigma_V) = (f(q_1, \sigma_1), f(q_2, \sigma_2));$$
- $q_{0,V} = (q_0, q_0)$ is the initial state.

Intuitively, V is constructed by synchronizing T with its own copy according to the observations under $\mathcal{H}_o$. Each state in V is a state pair in T and each event in V is also an event pair in T. The event set Σ_V is divided into Σ_V^o and Σ_V^{uo}, and (σ_1, σ_2) is in Σ_V^o iff $\sigma_1, \sigma_2 \in \Sigma_o$ and $\mathcal{H}_o(\sigma_1) = \mathcal{H}_o(\sigma_2)$. Also, (σ_1, σ_2) is in Σ_V^{uo} iff $\sigma_1(\sigma_2) \in \Sigma_{uo}$ and $\sigma_2(\sigma_1) = \epsilon$. By construction, the verification structure has following two properties. First, for any string $s = (s_1, s_2) \in \mathcal{L}(V)$, we have $\mathcal{H}_o(s_1) = \mathcal{H}_o(s_2)$. Second, for any strings $s_1, s_2 \in \mathcal{L}(G)$ such that $\mathcal{H}_o(s_1) = \mathcal{H}_o(s_2)$, there exists a string $s = (s_1, s_2)$ such that $s \in \mathcal{L}(V)$.

For the sake of convenience, for any $q_V = (q_1, q_2) \in Q_V$, we denote $\theta_1(q_V) = q_1$ and $\theta_2(q_V) = q_2$ as the first and second component of q_V, respectively. Now, we are ready to provide our main theorem.

Theorem 1. System G is not epistemically K-diagnosable w.r.t. $\mathcal{H}_o$, $\mathcal{H}_a$ and secret $X_S \subseteq X$ iff in the verification structure, there exists a string

$$q_{0,V} \xrightarrow{\sigma_V^1} q_{1,V} \xrightarrow{\sigma_V^2} \cdots \xrightarrow{\sigma_V^n} q_{n,V}$$

such that

1) $\theta_1(q_{i,V}) \in Q_R$ for some $i = 0, 1, ..., n$;
2) $\theta_1(q_{k,V}) \notin Q_R$ for all $k = 0, 1, ..., i-1$;
3) $\theta_2(q_{j,V}) \notin Q_R$ for all $j = 0, 1, ..., n$; and
4) $|\{\sigma_V^{i+1}, ..., \sigma_V^n\} \cap \{(\epsilon, \sigma_2) \in \Sigma_V^{uo}\}| \geq K$.

Note that in condition 3), we do not omit the repetitive elements in set $\{\sigma_V^{i+1}, ..., \sigma_V^n\}$. Then the above conditions can be understood as follows. For string $s = (s_1, s_2) = \sigma_V^1 \sigma_V^2 ... \sigma_V^n$, condition 1) and 2) ensures that, at some time instant i, the corresponding prefix of string s_1 can lead to information leakage. Condition 3) ensures that s_2, and all of its prefixes, do not lead to information leakage. Condition 4) says that, after the information is released, s_1 is still processed for at least K steps in the sense of non-empty events. By construction, s_1 and s_2 have the same observation from the user's point of view. Therefore, such a string essentially is a counterexample of epistemic diagnosability. We use the following example to illustrate the above results.

Example 3. We still consider the running example, and we only consider the verification of epistemic 2-diagnosability. Since we already show the secret recognizer in Figure 2, we can further construct the verification structure which is partially shown in Figure 3. Note that in this figure, without loss of generality and for the purpose of verification, we only show the part satisfying the conditions in Theorem 1 and omit other parts. Let us consider the string from $(1A, 1A)$ to $(6'E, 6A)$ and consider the part highlighted by red color, i.e.,

$$(4'D, 4A) \xrightarrow{(c', c)} (5'E, 5A) \xrightarrow{(a, a)} (6'E, 6A)$$

We notice that condition 1) and 2) is satisfied at state $(4'D, 4A)$ because $4'D \in Q_R$ and $1A, 1'B, 2'C, 3'C \notin$

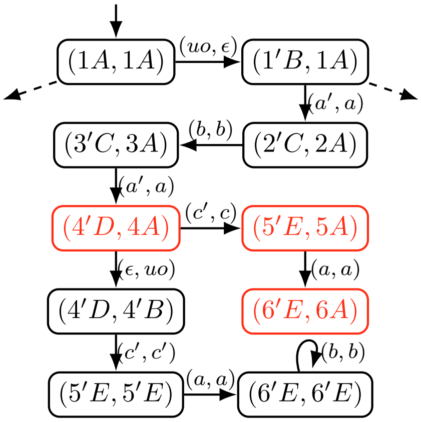

Fig. 3. Verification structure of the system shown in Fig. 1.

Q_R. Since all the states passed by another string, i.e., $1A, 2A, ..., 6A$, are not information leakage states, condition 3) is satisfied. Finally, in the highlighted part, we have $|\{(c'c), (a, a)\}| \geq 2$. Therefore, the system is not epistemically 2-diagnosable, which is consistent with our previous analysis in Example 1.

5. CONCLUSION

In this paper, we revisited the diagnosability analysis problem of DES from a new perspective. Specifically, in contrast to the standard diagnosability analysis, where faults are related to the internal behaviors of the system, we considered faults are some critical information leakages to an intruder with different observation. Then the notion of epistemic diagnosability was proposed to capture whether or not the system user can always detect the critical information leakage in time within a given delay bound. We argued that this new notion subsumes the notion of event-based diagnosability. We also provided an effective algorithm to verify this new notion. In the future, we would like to investigate the control synthesis problem to enforce epistemic diagnosability.

REFERENCES

Basile, F., Cabasino, M.P., and Seatzu, C. (2014). State estimation and fault diagnosis of labeled time Petri net systems with unobservable transitions. *IEEE Transactions on Automatic Control*, 60(4), 997–1009.

Basilio, J.C., Hadjicostis, C.N., Su, R., et al. (2021). Analysis and control for resilience of discrete event systems: Fault diagnosis, opacity and cyber security. *Foundations and Trends® in Systems and Control*, 8(4), 285–443.

Carvalho, L.K., Moreira, M.V., and Basilio, J.C. (2021). Comparative analysis of related notions of robust diagnosability of discrete-event systems. *Annual Reviews in Control*, 51, 23–36.

Cassandras, C.G. and Lafortune, S. (2008). *Introduction to discrete event systems*. Springer.

Chen, J. and Kumar, R. (2015). Fault detection of discrete-time stochastic systems subject to temporal logic correctness requirements. *IEEE Transactions on Automation Science and Engineering*, 12(4), 1369–1379.

Cui, B., Yin, X., Li, S., and Giua, A. (2022). You don't know what I know: On notion of high-order opacity in discrete-event systems. *IFAC-PapersOnLine*, 55(28), 135–141.

Dong, W., Li, S., and Yin, X. (2023a). Diagnosis of time-sensitive failures in timed discrete-event systems with metric interval temporal logics. In *2023 IEEE 62nd Conference on Decision and Control (CDC)*, 6821–6827. IEEE.

Dong, W., Yin, X., and Li, S. (2023b). A uniform framework for diagnosis of discrete-event systems with unreliable sensors using linear temporal logic. *IEEE Transactions on Automatic Control*.

Falcone, Y. and Marchand, H. (2015). Enforcement and validation (at runtime) of various notions of opacity. *Discrete Event Dynamic Systems*, 25, 531–570.

Jéron, T., Marchand, H., Pinchinat, S., and Cordier, M.O. (2006). Supervision patterns in discrete event systems diagnosis. In *2006 8th International Workshop on Discrete Event Systems*, 262–268. IEEE.

Jiang, S. and Kumar, R. (2004). Failure diagnosis of discrete-event systems with linear-time temporal logic specifications. *IEEE Transactions on Automatic Control*, 49(6), 934–945.

Lefebvre, D. and Delherm, C. (2007). Diagnosis of DES with Petri net models. *IEEE Transactions on Automation Science and Engineering*, 4(1), 114–118.

Liu, S., Trivedi, A., Yin, X., and Zamani, M. (2022). Secure-by-construction synthesis of cyber-physical systems. *Annual Reviews in Control*, 53, 30–50.

Ma, Z., Tong, Y., and Seatzu, C. (2023). Verification of pattern-pattern diagnosability in partially observed discrete event systems. *IEEE Transactions on Automatic Control*.

Ma, Z., Yin, X., and Li, Z. (2021). Verification and enforcement of strong infinite-and k-step opacity using state recognizers. *Automatica*, 133, 109838.

Pencolé, Y. and Subias, A. (2021). Diagnosability of event patterns in safe labeled time Petri nets: a model-checking approach. *IEEE Transactions on Automation Science and Engineering*, 19(2), 1151–1162.

Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K., and Teneketzis, D. (1995). Diagnosability of discrete-event systems. *IEEE Transactions on automatic control*, 40(9), 1555–1575.

Takai, S. (2021). A general framework for diagnosis of discrete event systems subject to sensor failures. *Automatica*, 129, 109669.

Van Ditmarsch, H., van Der Hoek, W., and Kooi, B. (2007). *Dynamic epistemic logic*, volume 337. Springer Science & Business Media.

Yin, X., Chen, J., Li, Z., and Li, S. (2019). Robust fault diagnosis of stochastic discrete event systems. *IEEE Transactions on Automatic Control*, 64(10), 4237–4244.

Yin, X. and Lafortune, S. (2017). On the decidability and complexity of diagnosability for labeled Petri nets. *IEEE Transactions on Automatic Control*, 62(11), 5931–5938.

Zad, S.H., Kwong, R.H., and Wonham, W.M. (2003). Fault diagnosis in discrete-event systems: Framework and model reduction. *IEEE Transactions on Automatic Control*, 48(7), 1199–1212.

Zhao, J., Yin, X., and Li, S. (2024). A unified framework for verification of observational properties for partially-observed discrete-event systems. *IEEE Transactions on Automatic Control*.