frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 17 April 2023
DOI 10.3389/fphy.2023.1185103

OPEN ACCESS

EDITED BY
Kaijie Xu,
University of Alberta, Canada

REVIEWED BY
Peng Nie,
Xidian University, China
Yihui Hu,
Xi'an University of Posts and Telecommunications, China
Jiazhong Zhou,
Huaqiao University, China

*CORRESPONDENCE
Qi Zhang,
zhangqi@ecut.edu.cn

SPECIALTY SECTION
This article was submitted to Optics and Photonics, a section of the journal Frontiers in Physics

RECEIVED 13 March 2023
ACCEPTED 31 March 2023
PUBLISHED 17 April 2023

CITATION
Zhang Q (2023), Robust predictability in discrete event systems under sensor attacks.
Front. Phys. 11:1185103.
doi: 10.3389/fphy.2023.1185103

Robust predictability in discrete event systems under sensor attacks

Qi Zhang*

School of Information Engineering, East China University of Technology, Nanchang, China

The problem of robust predictability against sensor attacks is investigated. The objective of a diagnoser is to predict the occurrence of a critical event of a discrete event system (DES) under partial observation. An attacker may rewrite the diagnoser observation by inserting fake events or erasing real events. Two novel structures, namely, real diagnoser and the fake diagnoser, are constructed based on the diagnoser of the system. We compute the hybrid diagnoser as the parallel composition of the real diagnoser and the fake diagnoser. The hybrid diagnoser can be used to verify if a critical event of the system is robustly predictable when an attacker tampers with the diagnoser observation.

KEYWORDS

discrete event system, automaton, predictability, diagnoser, sensor attack

1 Introduction

Suppose that a plant is modeled by a discrete event system (DES) under partial observation, *predictability* is a property that describes if a diagnoser can predict the occurrence of a critical event (either observable or unobservable) according to its observation of the system. As the system and the diagnoser are connected via a network, a malicious attacker may corrupt such a communication channel with the insertion of fake events and the deletion of real events that have happened in the system. Therefore, the problem of robust predictability against sensor attacks is addressed. It characterizes the ability of a diagnoser to predict the occurrence of a critical event, even if an attacker may tamper with its observation.

Genc and Lafortune [1] proposed the problem of predictability in the centralized case, and Kumar and Takai [2] considered this problem in the decentralized case. From this point, many studies have focused on this topic in different contexts and problem settings. Takai and Kumar [3, 4] considered the problem of failure prognosis with communication delays. In [5–7], the problem of predictability is studied in the context of stochastic DESs. Benmessahel et al. [8] investigated the problem of predictability in fuzzy DESs. Yin and Li [9] studied the problem of reliable decentralized fault predictability. They supposed that only partial local prognostic decisions are accessible to the coordinator. In [10], the authors showed how to use one prognoser to predict the occurrence of any failure for a set of models. Xiao and Liu [11] considered the problem of robust fault prognosis against loss of observations, where some observable events may become unobservable because of sensor failures. Finally, the problem of predictability is investigated in [12–14] in the framework of Petri nets.

The notion of diagnosability was first proposed in [15]. We assume that a DES contains an unobservable fault event. A fault event is said to be diagnosable if we can determine its occurrence within a limited delay. We point out that if the property of predictability is

stronger than that of diagnosability, i.e., if an event is predictable, then this event is also diagnosable.

The problem of robust codiagnosability against Denial-of-Service and deception attacks has been considered in [16]. The authors assume that an attacker can insert fake packages into the network that transmits the sensor readings such that delays and loss of observations may occur. They construct a new diagnoser to verify the property of robust codiagnosability. In [17], the problem of robust codiagnosability against sensor attacks under cost constraint is proposed. The considered attacks include symbol insertion, symbol erasure, and symbol replacement attacks. They assumed that each attack action consumes a certain amount of cost. They developed a strategy to verify the robust codiagnosability against an attacker with a bounded total cost.

Mainly inspired by [16, 17] that considered the problem of robust diagnosability in DESs subject to cyberattacks, we propose the problem of robust predictability in DESs subject to sensor attacks. To the best of the author's knowledge, this problem has not been considered in the framework of DESs. We finally mention that in [18], a structure named joint estimator is addressed to solve the problem of joint state estimation under attacks. This is a general structure that can be used to consider a set of problems in DESs subject to sensor attacks. In this work, we extend such a structure to solve the problem of robust predictability against sensor attacks.

In Section 2, the automata model and the notions of predictability and diagnoser are given. In Section 3, the problem considered in this study is presented. In Section 4, the real diagnoser is computed. It characterizes the real evolution of the diagnoser subject to sensor attacks. In Section 5, the fake diagnoser is constructed. It characterizes the fake evolution of the diagnoser subject to sensor attacks. In Section 6, the hybrid diagnoser is computed. It allows us to test if a critical event is robustly predictable. Section 7 summarizes the main results of this work, and the possible future work is also pointed out.

2 Preliminaries

Let E be an alphabet and L a language defined over E^*. The prefix closure of L is defined by $\bar{L} = \{\sigma \in E^* \mid (\exists \sigma' \in E^*)\ \sigma\sigma' \in L\}$. The post language of L after $\sigma \in L$ is defined as $L/\sigma = \{\sigma' \in E^* \mid \sigma\sigma' \in L\}$. A language L is live if for all $\sigma \in L$, there always exists $e \in E$ such that $\sigma e \in L$. The set of words in L that end with event f is defined by $\Psi(f, L) = \{\sigma f \in L \mid \sigma \in E^*, f \in E\}$.

A deterministic finite-state automaton (DFA), denoted by G, is a four tuple $G = \{X, E, \delta, x_0\}$, where X is a set of states; E is a finite set of events; δ: $X \times E \to X$ is the transition function and can be extended from the domain $X \times E$ to the domain $X \times E^*$, that is, $\delta(x, \varepsilon) := x$, and $\delta^*(x, \sigma e) := \delta(\delta^*(x, \sigma), e)$, where $e \in E$, $\sigma \in E^*$, and x_0 is the initial state. The generated language of G is defined by $L(G) = \{\sigma \in E^* \mid \delta^*(x, \sigma)$ is defined$\}$. The set of active events at state x of G is defined by $\Gamma_G(x) = \{e \in E \mid \delta(x, e)$ is defined$\}$.

A set of states $\{x_1, x_2, \ldots, x_n\} \subseteq X$ and a word $\sigma = e_1 e_2 \ldots e_n \in E^*$ form a cycle if $\delta(x_i, e_i) = x_{i+1}$, $i = 1, 2, \ldots, n-1$, and $\delta(x_n, e_n) = x_1$. The accessible part of G with respect to state x is defined as $Ac(G, x) = (X_{ac}, E, \delta_{ac}, x_0)$, where $X_{ac} = \{x' \in X \mid (\exists \sigma \in E^*)\ \delta^*(x, \sigma) = x'\}$, $\delta_{ac} = \delta|_{X_{ac} \times E \to X_{ac}}$.

Due to the lack of observability in the system, E is divided into the set of observable events E_o and the set of unobservable events E_{uo}. The natural projection on E_o is denoted as P: $E^* \to E_o^*$. Considering a word $\sigma \in E^*$, $P(\sigma)$ simply removes the unobservable events from σ, that is, $P(\varepsilon) := \varepsilon$ and $P(\sigma e) := P(\sigma)e$ if $e \in E_o$ and $P(\sigma e) := P(\sigma)$ if $e \in E \setminus E_o$.

Definition 1. [1] Consider a prefix-closed and live language L on alphabet E. An event f is said to be predictable with respect to P if $(\exists n \in \mathbb{N})\ \forall \sigma \in \Psi(f, L), \exists t \in \bar{\sigma}$ such that $f \notin t \wedge \mathcal{P}$, where condition $\mathcal{P}$:

$\forall u \in L$ such that $P(u) = P(t)$, $f \notin u$. $\forall v \in L/u$ such that $|v| \geq n \Rightarrow f \in v$.

In plain words, an event f is predictable if it holds that once the observation $P(t)$ is produced, f will necessarily occur within n steps, where t is a normal prefix of a word σ that ends with f.

Definition 2. [1] Let $G = (X, E, \delta, x_0)$ be a plant and f an event that needs to be predicted. The diagnoser is a DFA, denoted as $D_g = (B, E_o, \delta_d, b_0)$, where

- $B \subseteq 2^{X \times \{N,F\}}$, for example, $b = \{(x_1, l_1), \ldots, (x_n, l_n)\}$, and $x_1, x_2 \ldots, x_n \in X$;
- δ_d: $B \times E_o \to B$, for example, if $\exists e \in E_o$ such that $\delta_d(b, e) = b'$, where $b = \{(x_1, l_1), \ldots, (x_m, l_m)\}$ and $b' = \{(x_1', l_1'), \ldots, (x_n', l_n')\}$, then $\exists i \in \{1, \ldots, m\}$, $\exists j \in \{1, \ldots, n\}$, and $\exists \sigma = te$: $t \in E_{uo}^*$ such that $\delta^*(x_i, \sigma) = x_j'$, where

$$l_j' = \begin{cases} N & if \quad l_i = N \wedge f \notin \sigma, \\ F & if \quad l_i = F \vee f \in \sigma. \end{cases}$$

If a state of the diagnoser is labeled N, it indicates that event f has not happened when the current state is reached. If a state of the diagnoser is labeled F, it implies that event f has happened when the current state is reached. By convention, the unobservable reach is not included in a diagnoser state.

Definition 3. [1] In the diagnoser $D_g = (B, E_o, \delta_d, b_0)$,

- We define $B_n = \{b = \{(x_1, l_1), \ldots, (x_n, l_n)\} \in B \mid \forall\, l_i \in \{l_1, \ldots, l_n\}, l_i = N\}$ as the set of normal states of D_g.
- We define $B_c = \{b = \{(x_1, l_1), \ldots, (x_n, l_n)\} \in B \mid \forall\, l_i \in \{l_1, \ldots, l_n\}, l_i = F\}$ as the set of certain states of D_g.
- We define $B_{uc} = \{b = \{(x_1, l_1), \ldots, (x_n, l_n)\} \in B \mid \exists\, l_i, l_j \in \{l_1, \ldots, l_n\}, l_i = N, l_j = F\}$ as the set of uncertain states of D_g.
- We denote by B_d the set of normal states with an instantaneous continuator, which is not normal, that is, $B_d = \{b \in B_n \mid (\exists e \in E_o)\ \delta_d(b, e) \notin B_n\}$.

In other words, a state $b \in B$ is normal if all the labels within it are N; a state $b \in B$ is certain if all the labels within it are F; and a state $b \in B$ is uncertain if there exist labels N and F within it.

Theorem 4. [1] Let G be a plant and $D_g = (B, E_o, \delta_d, b_0)$ its diagnoser. An event f is predictable if and only if for all $b_d \in B_d$, in the accessible part of the diagnoser $Ac(D_g, b)$, all cycles are cycles of certain states.

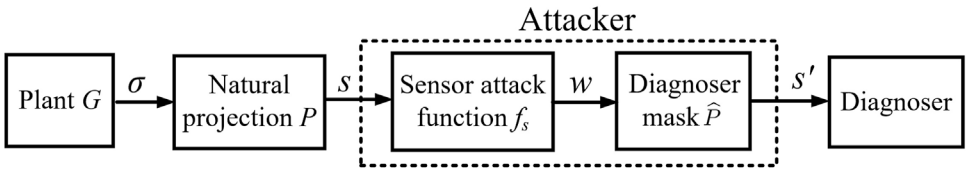

FIGURE 1
System under attack.

3 Problem formulation

Let $G = (X, E, \delta, x_0)$ be a plant modeled by a DFA. As shown in Figure 1, if the word $\sigma \in E^\star$ is generated by G, the observation $s = P(\sigma)$ may be corrupted by an attacker. Then, the diagnoser predicts the occurrence of a critical event in accordance with the corrupted observation s'. It should be noted that the internal structure of the attacker within the dotted lines will be discussed later.

Suppose that an attacker can only tamper with a subset of events of G, we call this subset the set of compromised events E_{com}. We divide E_{com} into two subsets, that is, $E_{com} = E_{ins} \cup E_{era}$, where E_{ins} is the set of events that may be inserted into the diagnoser observation, and E_{era} is the set of events that may be deleted from the diagnoser observation. The sets E_{ins} and E_{era} may contain common events.

To make a distinction between the attacker's action from the original behavior of G, we define two new sets of events. We denote by E_+ the set of inserted events, defined as $E_+ = \{e_+ \mid e \in E_{ins}\}$ [19]. We denote by E_- the set of erased events, defined as $E_- = \{e_- \mid e \in E_{era}\}$ [19]. If $e_+ \in E_+$ happens, it indicates that an attacker inserts the fake symbol $e \in E_{ins}$ into the diagnoser observation. If $e_- \in E_-$ happens, it indicates that an attacker erases the real symbol $e \in E_{era}$ from the diagnoser observation. Finally, we denote by E_a the attack alphabet, defined as $E_a = E_o \cup E_+ \cup E_-$. We point out that the three subsets E_o, E_+, and E_- are disjoint.

Definition 5. Let G be a plant and $E_{com} = E_{ins} \cup E_{era}$ the set of compromised events. An attacker is defined by a sensor attack function f_s: $P[L(G)] \to E_a^\star$:

(1) $f_s(\varepsilon) \in E_+^\star$,
(2) $\forall se \in P[L(G)]$:

$$\begin{cases} f_s(se) \in f_s(s)\{e_-, e\}E_+^\star & if \quad e \in E_{era}, \\ f_s(se) \in f_s(s)eE_+^\star & if \quad e \in E_o \backslash E_{era}. \end{cases} \tag{1}$$

Condition (1) means that a word in $E_+^\star$ can be inserted by the attacker before an observable event occurs in G. Condition (2) means that when an event that can be erased by the attacker occurs, the attacker either erases it or not; then, it inserts any word defined over $E_+^\star$. Finally, when an event that cannot be erased by the attacker happens, the attacker can insert a word defined over $E_+^\star$ after it.

Let G be a plant. We denote by $L(f_s, G)$ the attack language, defined by $L(f_s, G) = f_s(P[L(G)]) \subseteq E_a^\star$. We call $w \in L(f_s, G)$ an attack word. We denote by $\mathcal{F}_s$ the set of sensor attack functions. We denote by $L(\mathcal{F}_s, G)$ the union of all the attack languages, defined by $L(\mathcal{F}_s, G) = \bigcup_{f_s \in \mathcal{F}_s} f_s(P[L(G)])$.

Definition 6. The real mask $\tilde{P}$: $E_a^\star \to E_o^\star$ is defined as follows:

$$\tilde{P}(\varepsilon) = \varepsilon, \ \tilde{P}(we') = \begin{cases} \tilde{P}(w)e \ if \ e' = e \in E_o \vee e' = e_- \in E_-, \\ \tilde{P}(w) \ \ if \ e' = e_+ \in E_+. \end{cases} \tag{2}$$

In plain words, the real mask transforms events in E_a into real events that have happened in the system. As e_- means an erased event that has happened in the system, e_- is transformed into the corresponding event $e \in E_o$. e_+ is neglected because it is a fake event.

Definition 7. The diagnoser mask $\hat{P}$: $E_a^\star \to E_o^\star$ is defined as follows:

$$\hat{P}(\varepsilon) = \varepsilon, \ \hat{P}(we') = \begin{cases} \hat{P}(w)e \ if \ e' = e \in E_o \vee e' = e_+ \in E_+, \\ \hat{P}(w) \ \ if \ e' = e_- \in E_-. \end{cases} \tag{3}$$

In simple words, the diagnoser mask characterizes how the diagnoser observes events in E_a. Namely, the diagnoser cannot distinguish the real event $e \in E_o$ from the inserted event $e_+ \in E_+$, and it cannot observe erased events in E_-.

As shown in Figure 1 within the dotted lines, the observation $s \in E_o$ is corrupted into the attack word $w \in E_a^\star$ by the sensor attack function f_s; then, w is transformed into the corrupt observation $s' = \hat{P}(w)$. Therefore, the diagnoser actually observes s'.

In this study, let G be a plant. The following two assumptions are made:

1) The generated language $L(G)$ is live.
2) In G, there does not exist a cycle that consists of unobservable events only.

Assumption 1) is made for the sake of simplicity. Assumption 2) guarantees that plant G does not generate unobservable words with infinite length.

Definition 8. Let G be a plant that satisfies Assumption 1) and Assumption 2). An event f is robustly predictable with respect to P if $(\exists n \in \mathbb{N})\ \forall \sigma \in \Psi(f, L),\ \exists t \in \bar{\sigma}$ such that $f \notin t \wedge \mathcal{P}_r$, where condition $\mathcal{P}_r$:

$\forall w \in L(\mathcal{F}_s, G)$ such that $\tilde{P}(w) = P(t) \vee \hat{P}(w) = P(t)$. $\forall u \in L(G)$ such that $P(u) = \tilde{P}(w) \vee P(u) = \hat{P}(w)$, $f \notin u$. $\forall v \in L(G)/u$ such that $|v| \geq n \Rightarrow f \in v$.

In Definition 8, let t be a normal prefix of a word σ that ends with f. We use t to find all the attack words $w \in E_a^\star$ such that $\tilde{P}(w) = P(t) \vee \hat{P}(w) = P(t)$. Then, we use these attack words w to find all the word $u \in E^\star$ such that $P(u) = \tilde{P}(w) \vee P(u) = \hat{P}(w)$.

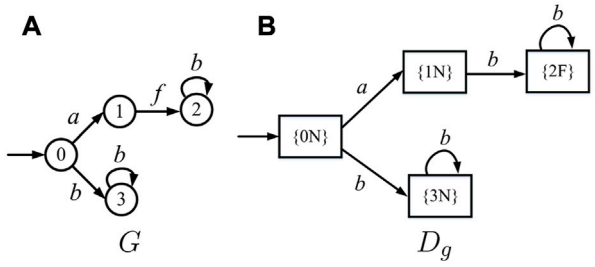

FIGURE 2
(A) Plant G and **(B)** diagnoser D_g in Example 10.

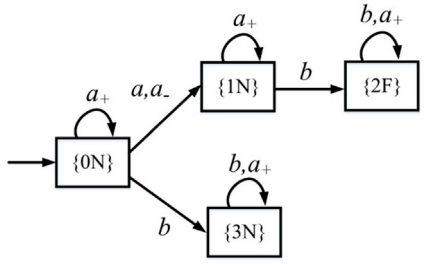

FIGURE 3
Real diagnoser D_r in Example 10.

An event f is robustly predictable if it holds that once the observation $P(u)$ is produced, then f will necessarily occur within n steps.

We point out that, for each attack word w, we distinguish the observations $P(u) = \tilde{P}(w)$ and $P(u) = \hat{P}(w)$ because the attacker can make these two observations look alike for the diagnoser.

4 Real diagnoser

The real diagnoser D_r describes the real evolution of the diagnoser in accordance with the attack alphabet E_a. Namely, the real diagnoser changes its states the same way in terms of $e \in E_{era}$ and the corresponding events e_-; the real diagnoser does not change its states when the fake event $e_+ \in E_+$ happens.

Definition 9. Let $G = (X, E, \delta, x_0)$ be a plant and $D_g = (B, E_o, \delta_d, b_0)$ the diagnoser. The real diagnoser is a DFA $D_r = (B, E_a, \delta_r, b_0)$, and its transition function δ_r satisfies the following:

$$\begin{cases} for\ all\ b \in B,\ for\ all\ e \in E_o: & \delta_r(b,e) \coloneqq \delta_d(b,e), \\ for\ all\ b \in B,\ for\ all\ e \in E_{era}: & \delta_r(b,e_-) \coloneqq \delta_r(b,e), \\ for\ all\ b \in B,\ for\ all\ e \in E_{ins}: & \delta_r(b,e_+) \coloneqq b. \end{cases} \tag{4}$$

The construction of the real diagnoser can be explained as follows: first, we set the transition function of the real diagnoser D_r equal to the transition function of the diagnoser D_g. Then, each time there is a transition labeled $e \in E_{era}$, we add a transition labeled e_-. Finally, for each event in E_{ins}, for each state of D_r, we add a self-loop labeled e_+.

We point out that the real diagnoser D_r is similar to the attacker observer constructed by Algorithm 1 in [18]. Although the input of Algorithm 1 is the observer of G, here we replace it with the diagnoser of G.

Example 10. As sketched in Figure 2A, let G be the plant, $E_o = \{a, b\}$, and $E_{uo} = \{f\}$. Assume that f is the event that needs to be predicted. The diagnoser $D_g = (B, E_o, \delta_d, b_0)$ is sketched in Figure 2B.

Let $E_{ins} = E_{era} = \{a\}$. The real diagnoser is shown in Figure 3. We add a transition $\delta_r(\{0N\}, a_-) = \{1N\}$ in D_r because there exists a transition $\delta_d(\{0N\}, a) = \{1N\}$ such that $e \in E_{era}$ in D_g. Self-loops labeled a_+ are added at all the states of D_r because $a \in E_{ins}$.

Proposition 11. Let G be the plant, $D_g = (B, E_a, \delta, b_0)$ its diagnoser, and $D_r = (B, E_a, \delta_r, b_0)$ the real diagnoser.

(i) $L(D_r) = L(\mathcal{F}_s, G)$;

(ii) $\forall s\ L(D_g), \forall f_s \in \mathcal{F}_s$ with $w = f_s(s) \in E_a^*$: $\delta_r^*(b_0, w) = \delta_d^*(b_0, s)$.

Proof. The proof is neglected because it is the same as the proof of Proposition 1 in [18]. In simple words, item 1) means that the real diagnoser generates the union of all the attack languages. Item 2) indicates that the state arrived in D_r by implementing $w = f_s(s) \in E_a^*$ equal to the state arrived in D_g by implementing $s \in E_o^*$.

5 Fake diagnoser

The fake diagnoser D_f describes the fake evolution of the diagnoser in accordance with the attack alphabet E_a. Namely, the fake diagnoser changes its states the same way in terms of $e \in E_{ins}$ and the corresponding events e_+ because it cannot distinguish the real event of the plant e from the fake event e_+. The fake diagnoser does not change its states in case of the occurrence of $e_- \in E_-$ because it cannot observe the erased event e_-. We add a new state $b_\varnothing$ in D_r. The diagnoser knows that the plant is under attack when this state is reached.

Definition 12. Let $G = (X, E, \delta, x_0)$ be a plant and $D_g = (B, E_o, \delta_d, b_0)$ the diagnoser. The fake diagnoser is a DFA $D_f = (B_f, E_a, \delta_f, b_0)$ such that $B_f = B \cup b_\varnothing$, and its transition function δ_f satisfies the following:

$$\begin{cases} for\ all\ b \in B,\ for\ all\ e \in E_o: & \delta_f(b,e) \coloneqq \delta_d(b,e), \\ for\ all\ b \in B,\ for\ all\ e \in E_{ins}: & \delta_f(b,e_+) \coloneqq \delta_f(b,e), \\ for\ all\ b \in B,\ for\ all\ e \in E_{era}: & \delta_f(b,e_-) \coloneqq b, \\ for\ all\ b \in B,\ for\ all\ e \in E_a: & if\ \delta_f(b,e)\ is\ undefined,\ then\ \delta_f(b,e) \coloneqq b_\varnothing. \end{cases} \tag{5}$$

The construction of the fake diagnoser can be explained as follows: first, we set the transition function of D_f equal to the transition function of the diagnoser D_g. Then, each time there is a transition labeled $e \in E_{ins}$, we add a transition labeled $e_+ \in E_+$. Self-loop labeled events in E_- are added at all the states of D_f. Finally, for each event in E_a and each state in B, we set $\delta_f(b, e_a) = b_\varnothing$ for all the undefined transitions. Note that state $b_\varnothing$ has no input and output arcs.

We point out that the fake diagnoser D_f is similar to the operator observer computed by Algorithm 2 of [18]. Although the input of Algorithm 2 is the observer of G, here we replace it with the diagnoser of G.

Example 13. Recall plant G with its diagnoser D_g in Example 10. Suppose that $E_{ins} = E_{era} = \{a\}$. Figure 4 shows the fake diagnoser.

First, we add a transition $\delta_r(\{0N\}, a_+) = \{1N\}$ in D_f as there is a transition $\delta_d(\{0N\}, a) = \{1N\}$ such that $e \in E_{ins}$ in D_g. Then, for all the states of D_f, self-loops labeled a_- are added because $a \in E_{era}$. Finally, all the undefined transitions lead to the state $b_\varnothing$.

The following definitions are given to formalize the generated language of the fake diagnoser D_f.

Definition 14. Consider a plant G with the fake diagnoser D_f.

- A sensor attack function f_s is stealthy if $\hat{P}[L(f_s, G)] \subseteq P[L(G)]$.
- The set of stealthy words is defined as $W_s = \{w \in E_a^* \mid \hat{P}(w) \in P[L(G)]\}$.
- The set of exposing words is defined as $W_e = \{we_a \in E_a^* \mid w \in W_s, e_a \in E_a, we_a \notin W_s\}$.

According to Definition 14, f_s is stealthy if the attack words in $L(f_s, G) \subseteq E_a^*$ can be transformed into words in $P[L(G)] \subseteq E_o^*$ via the diagnoser mask $\hat{P}$; that is, the diagnoser cannot discover the presence of an attacker. Set W_s includes all the words that keep the attacker stealthy. Each word in W_e is the concatenation of a stealthy word and an event in E_a, and the resulting word is no more stealthy.

Proposition 15. Let G be the plant, $D_g = (B, E_o, \delta_d, b_0)$ the diagnoser, and $D_f = (B, E_a, \delta_f, b_0)$ the fake diagnoser.

(i) $L(D_f) = W_s \cup W_e$;
(ii) $\forall w \in L(D_f)$: if $w \in W_s$, then $\delta_f^*(b_0, w) = \delta_d^*(b_0, \hat{P}(w))$; if $w \in W_e$, then $\delta_f^*(b_0, w) = b_\varnothing$.

Proof. The proof is ignored because it is the same as the proof of Proposition 2 in [18]. In plain words, item (i) implies that the language of the fake diagnoser equals the union of W_s and W_e. Item (ii) means that the state arrived in D_f by implementing $w \in E_a^*$ equal to the state arrived in D_g by implementing $\hat{P}(w) \in E_o^*$, and all the exposing words lead to state $b_\varnothing$.

6 Hybrid diagnoser

The notion of the hybrid diagnoser is given on the basis of the real diagnoser and fake diagnoser.

Definition 16. Let $G = (X, E, \delta, x_0)$ be a plant, $D_r = (B, E_a, \delta_r, b_0)$ the real diagnoser, and $D_f = (B_f, E_a, \delta_f, b_0)$ the fake diagnoser. The hybrid diagnoser $D_h = (R, E_a, \delta_h, r_0)$ is defined as the parallel composition of D_r and D_h, that is, $D_h = D_r \| D_f$, where

- $R = (b, b_f) \subseteq 2^{X\times\{N,F\}} \times 2^{X\times\{N,F\}}$;
- $\delta_h[(b, b_f), e] = [\delta_r(b, e), \delta_f(b_f, e)]$ if $e \in \Gamma_{D_r}(b) \cap \Gamma_{D_f}(b_f)$, where $\Gamma_{D_r}(b)$ $(\Gamma_{D_f}(b_f))$ denotes the set of active events at state b (b_f) of D_r (D_f);
- the initial state is $r_0 = (b_0, b_0)$.

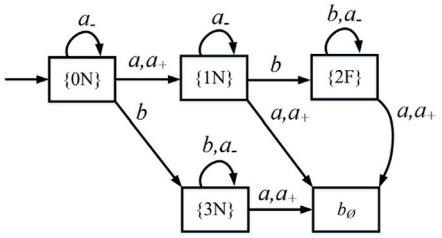

FIGURE 4
Fake diagnoser D_f in Example 13.

Now, we investigate the complexity of building the hybrid diagnoser D_h. Let $G = (X, E, \delta, x_0)$ be a plant. Its diagnoser D_g is built in $2^{|X|}$ steps. In accordance with Definition 9, the real diagnoser D_r contains at most $2^{|X|}$ states. In accordance with Definition 12, the fake diagnoser D_f contains at most $2^{|X|} + 1$ states. As $D_h = D_r \| D_f$, the computational complexity to build D_h is $O(2^{|X|} \cdot 2^{|X|})$.

Example 17. Recall plant G in Example 10. The hybrid diagnoser $D_h = D_r \| D_f$ is sketched in Figure 5, where D_r (D_f) is sketched in Figure 3 (Figure 4).

Definition 18. Let G be the plant, and $D_h = (R, E_a, \delta_h, r_0)$ be the hybrid diagnoser:

- We define $R_n = \{r = (b, b_f) \in R \mid b = \{(x_1, l_1), \ldots, (x_m, l_m)\}, b_f = \{(x_1', l_1'), \ldots, (x_n', l_n')\}$ such that $\forall\, l_i \in \{l_1, \ldots, l_m\}, \forall\, l_j' \in \{l_1', \ldots, l_n'\}, l_i = N, l_j' = N\}$ the set of normal states of D_h.
- We define $R_c = \{r = (b, b_f) \in R \mid b = \{(x_1, l_1), \ldots, (x_m, l_m)\}, b_f = \{(x_1', l_1'), \ldots, (x_n', l_n')\}$ such that $\forall\, l_i \in \{l_1, \ldots, l_m\}, \forall\, l_j' \in \{l_1', \ldots, l_n'\}, l_i = F, l_j' = F\}$ the set of certain states of D_h.
- We define $R_{uc} = \{r = (b, b_f) \in R \mid b = \{(x_1, l_1), \ldots, (x_m, l_m)\}, b_f = \{(x_1', l_1'), \ldots, (x_n', l_n')\}$ such that $\exists\, l_i \in \{l_1, \ldots, l_m\}, \exists l_j' \in \{l_1', \ldots, l_n'\}, l_i = N$ $(resp.,\ F), l_j' = F$ $(resp.,\ N)\}$ the set of uncertain states of D_h.
- We denote by R_d the set of normal states with an instantaneous continuator, which is not normal, that is, $R_d = \{r \in R_n \mid (\exists e_a \in E_a)\ \delta_h(r, e_a) \notin R_n\}$.

We point out that Definition 18, defined in hybrid diagnoser D_h, is the counterpart of Definition 3, defined in the diagnoser D_g.

Theorem 19. Let G be a plant, $D_g = (B, E_o, \delta_d, b_0)$ the diagnoser, and $D_h = (R, E_a, \delta_h, r_0)$ the hybrid diagnoser.

(a) $L(D_h) = L(\mathcal{F}_s, G) \cap (W_s \cup W_e)$;
(b) $\forall s \in P[L(G)], \forall f_s \in \mathcal{F}_s$ with $w = f_s(s) \in E_a^*$;
  (i) If $w \in W_s$, then $\delta_h^*(r_0, w) = (b, b_f) \Leftrightarrow \delta_d^*(b_0, s) = b,\ \delta_d^*[b_0, \hat{P}(w)] = b_f$;
  (ii) If $w \in W_e$, then $\delta_h^*(r_0, w) = (b, b_\varnothing) \Leftrightarrow \delta_d^*(b_0, s) = b,$ $\delta_d^*[b_0, \hat{P}(w)]$ is undefined.

Proof. The proof is neglected because it is the same as the proof of Theorem 1 in [18]. In other words, item (a) implies that the language

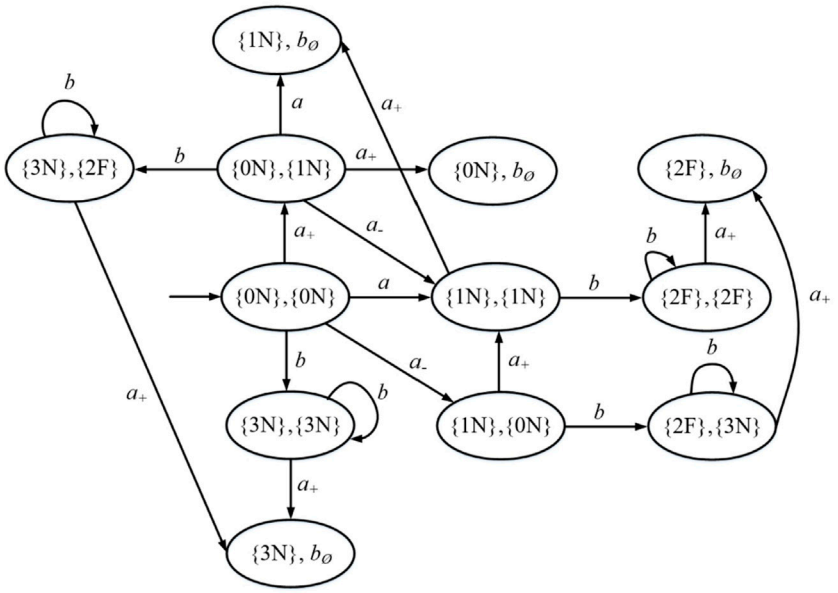

FIGURE 5
Hybrid diagnoser D_h in Example 17.

of the hybrid diagnoser D_h equals the intersection of the language of the real diagnoser and the language of the fake diagnoser.

Item (b) means that (i) if $w \in W_s$ and the state (b, b_f) is arrived in D_h by implementing $w = f_s(s)$, then the first element of this state equals the state arrived in the diagnoser D_g by implementing $s \in E_o^\star$. The second element of this state equals the state arrived in D_g by implementing $\hat{P}(w)$. (ii) If $w \in W_e$, then $\delta_d^\star(b_0, \hat{P}(w))$ is undefined.

Proposition 20. Let G be a plant and $D_h = (R, E_a, \delta_h, r_0)$ the hybrid diagnoser. In D_h, we suppose that a set of states $\{r_1, r_2, \ldots, r_n\} \subseteq R$ and a word $w = e_{a1}e_{a2}\ldots e_{an} \in E_a^\star$ form a cycle. If $\exists r_i \in R_c$, then $\forall r_j \in R_c$, where $i, j \in \{1, 2, \ldots, n\}$ and R_c are the set of certain states.

Proof. Proposition 20 means that in a cycle of D_h, if a certain state exists, then all the other states in this cycle are certain. The proof follows from the fact that the label F propagates; once a state is labeled as a certain state, all the states that are reachable from this state are also certain.

Proposition 21. Let G be a plant, $D_g = (B, E_o, \delta_d, b_0)$ the diagnoser, and $D_h = (R, E_a, \delta_h, r_0)$ the hybrid diagnoser. In D_h, if a set of states $\{(b_1, b_{f1}), (b_2, b_{f2}), \ldots, (b_n, b_{fn})\} \subseteq R$ and a word $w = e_{a1}e_{a2}\ldots e_{an} \in E_a^\star$ form a cycle, where $\forall i \in \{1, 2, \ldots, n\}$, $(b_i, b_{fi}) \in \{R_n \cup R_{uc}\}$. Then, in G, there exists a set of states $\{x_1, x_2, \ldots, x_n\} \subseteq X$ and a word $\sigma = e_1e_2\ldots e_n \in E^\star$ forming a cycle such that $\forall i \in \{1, 2, \ldots, n\}$, $(x_i, l_i) \in b_i$, $l_i = N$, $w = f_s[P(\sigma)]$ or $\forall i \in \{1, 2, \ldots, n\}$, $(x_i, l_i) \in b_{fi}$, $l_i = N$, $P(\sigma) = \hat{P}(w)$, where f_s is the sensor attack function, and $\hat{P}$ is the diagnoser mask.

Proof. Assume that, in the hybrid diagnoser D_h, a set of states $\{(b_1, b_{f1}), (b_2, b_{f2}), \ldots, (b_n, b_{fn})\} \subseteq R$ and a word $w = e_{a1}e_{a2}\ldots e_{an} \in E_a^\star$ form a cycle, where $\forall i \in \{1, 2, \ldots, n\}$, $(b_i, b_{fi}) \in \{R_n \cup R_{uc}\}$.

As $D_h = D_r \| D_f$, a set of states $\{b_1, , b_2, \ldots, b_n\} \subseteq B$ and the word $w = e_{a1}e_{a2}\ldots e_{an} \in E_a^\star$ form a cycle in the real diagnoser D_r, and a set of states $\{b_{f1}, , b_{f2}, \ldots, b_{fn}\} \subseteq B_f$ and the word $w = e_{a1}e_{a2}\ldots e_{an} \in E_a^\star$ form a cycle in the fake diagnoser D_f.

In accordance with Theorem 19, if $w \in W_s$, then $\delta_h^\star(r_0, w) = (b, b_f) \Leftrightarrow \delta_d^\star(b_0, s) = b$, $\delta_d^\star[b_0, \hat{P}(w)] = b_f$, where $w = f_s(s)$, $s = P(\sigma) \in E_o^\star$, and $\sigma = e_1e_2\ldots e_n \in E^\star$. As $\forall i \in \{1, 2, \ldots, n\}$, $(b_i, b_{fi}) \in \{R_n \cup R_{uc}\}$, we distinguish two cases: 1) If $\forall i \in \{1, 2, \ldots, n\}$, $(x_i, l_i) \in b_i$, $l_i = N$, then in G, a set of states $\{x_1, , x_2, \ldots, x_n\} \subseteq X$ and a word $\sigma = e_1e_2\ldots e_n \in E^\star$ form a cycle, where $w = f_s[P(\sigma)]$. 2) If $\forall i \in \{1, 2, \ldots, n\}$, $(x_i, l_i) \in b_{fi}$ and $l_i = N$, then in G, a set of states $\{x_1, , x_2, \ldots, x_n\} \subseteq X$ and a word $\sigma = e_1e_2\ldots e_n \in E^\star$ form a cycle, where $P(\sigma) = \hat{P}(w)$.

Note that as state $b_\varnothing$ has no output arcs in the fake diagnoser D_f, then in D_h, the cycle does not contain the state whose second element is $b_\varnothing$. Therefore, the case of $w \in W_e$ is not considered when we use the results of Theorem 19. For the same reason, we exclude this case in the proof of Theorem 22.

Theorem 22. Let $G = (X, E, \delta, x_0)$ be a plant and $D_h = (R, E_a, \delta_h, r_0)$ the hybrid diagnoser. An event f is robustly predictable if and only if, for all $r_d \in R_d$, in the accessible part of the hybrid diagnoser $Ac(D_h, r_d)$, all cycles are cycles of certain states.

Proof. (If) Assume that for all $r_d \in R_d$, in $Ac(D_h, r_d)$, all cycles are cycles of certain states. Consider a word $\sigma \in \Psi(f, L(G))$ such that $\delta^\star(x_0, \sigma) = x$. Let $\sigma_{uo}e_o \in L/\sigma$ such that $e_o \in E_o$ and $\delta^\star(x, \sigma_{uo}e_o) = x'$.

Consider a word w such that $\tilde{P}(w) = P(\sigma)$ or $\hat{P}(w) = P(\sigma)$. Let $\delta_h^\star(r_0, w) = r = (b, f_f)$ and $\delta_h(r, e_o) = r' = (b', b_f')$. According to

Theorem 19, $\delta_h^\star(r_0, we_o) = (b', b_f') = r' \Leftrightarrow \delta_d^\natural(b_0, s) = b'$, $\delta_d^\natural[b_0, \hat{P}(we_o)] = b_f'$. We consider the following two cases:

a) If $\tilde{P}(w) = P(\sigma)$, then $s = \tilde{P}(we_o) = P(\sigma e_o)$. It can be concluded that there exists $(x, l) \in b'$ such that $l = F$.
b) If $\hat{P}(w) = P(\sigma)$, then $\hat{P}(we_o) = P(\sigma e_o)$. It can be concluded that there exists $(x, l) \in b_f'$ such that $l = F$.

In any case, we can conclude that $r' \in R_{uc} \cup R_c$. As $\delta_h(r, e_o) = r'$, the following two cases are possible:

1) If $r \in R_n$, it means that $r \in R_d$ because $\delta_h(r, e_o) = r' \in \{R_{uc} \cup R_c\}$. Let $\sigma = tf$, where $t \in E^\star$. $\forall u \in L(G)$ such that $P(u) = \tilde{P}(w)$ or $P(u) = \hat{P}(w)$. As $\forall r_d \in R_d$, in $Ac(D_h, r_d)$, all cycles are cycles of certain states; then $\forall v \in L(G)/u$, $|v| \geq n$, and v contains f.
2) If $r \in R_{uc} \cup R_c$, then we can always find a state $r'' \in R_d$ such that state r is reachable from state r''. As a result, the proof for case 2) is reduced to the proof for case 1) by replacing r with r''.

(Only if) Assume that event f is robustly predictable, and there exists $r_d \in R_d$ such that $Ac(D_h, r_d)$ has a cycle that contains a state that is uncertain.

According to Proposition 20, in $Ac(D_h, r_d)$, as there exists a state that is uncertain in the cycle, then none of the states is certain in this cycle. In accordance with Proposition 21, as there exists a cycle where all the states are uncertain in $Ac(D_h, r_d)$, there exists a cycle where all the states are labeled N in plant G.

Suppose that, in D_h, $\delta_h^\star(r_0, w) = r_d = (b, b_f) \in R_d$. By Theorem 19, $\delta_h^\star(r_0, w) = (b, b_f) \Leftrightarrow \delta_d^\natural(b_0, s) = b$, $\delta_d^\natural[b_0, \hat{P}(w)] = b_f$. As $r_d \in R_d$, then there exists a word $\sigma \in \Psi(f, L(G))$ such that $\sigma = tf$, $t \in E^\star$, $\tilde{P}(w) = P(t)$ or $\hat{P}(w) = P(t)$. Let $r_1 = (b, b_f) \in R$ be a state of the cycle of $Ac(D_h, r_d)$ such that $\delta_h^\star(r_d, w') = r_1$. As $\delta_h^\star(r_0, w) = r_d$, then $\delta_h^\star(r_0, ww') = r_1$. Let x be a state of the cycle of G such that $\delta^\star(x_0, uv) = x$, and $\delta^\star(x, (e_1e_2 \ldots e_n)^m) = x$, where $u \in L(G)$, $v \in L(G)/u$ such that $P(u) = \tilde{P}(w)$ or $P(u) = \hat{P}(w)$. Then, $\delta^\star(x_0, uv(e_1e_2 \ldots e_n)^m) = x$. Because x is labeled by N in $Ac(D_h, r_d)$, then we can always find a word $v(e_1e_2 \ldots e_n)^m$ that does not contain f, and its length is greater than any $n \in \mathbb{N}$. As a result, the robustly predictable condition is violated, leading to a contradiction.

Example 23. Recall plant G in Example 10, where $E_o = \{a, b\}$ and $E_{uo} = \{f\}$. Assume that event f needs to be predicted. Let $E_{ins} = \{a\}$ and $E_{era} = \{a\}$.

In the diagnoser D_g in Figure 2B, state $\{1N\} \in B_d$. As $Ac(D, \{1N\})$ only contains one cycle (self-loop) labeled b at state $\{2F\}$, that is a certain state, according to Theorem 4, event f is predictable when no attack occurs.

In the hybrid diagnoser D_h visualized in Figure 5, states $(\{0N\}, \{1N\})$, $(\{1N\}, \{0N\})$, $(\{1N\}, \{1N\}) \in R_d$. As $Ac[D_h, (\{0N\}, \{1N\})]$ includes a cycle labeled b at state $(\{3N\}, \{2F\})$, that is not a certain state, and $Ac[D_h, (\{1N\}, \{0N\})]$ contains a cycle labeled b at state $(\{2F\}, \{3N\})$, that is not a certain state, in accordance with Theorem 22, event f is not robustly predictable when the attack occurs.

7 Conclusion

We consider the problem of robust predictability against sensor attacks. Based on a novel structure called hybrid diagnoser, an approach to test robust predictability is provided.

In the future, on one hand, as the construction of the diagnoser has exponential complexity, we intend to construct a verifier, which has polynomial complexity, to test robust predictability. On the other hand, we will try to extend the approach proposed in this work to the decentralized case.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

Author contributions

QZ writes the manuscript. The author agrees to be accountable for the content of the work.

Funding

This work was supported by the Scientific Research Startup Fund of East China University of Technology.

Conflict of interest

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

1. Genc S, Lafortune S. Predictability of event occurrences in partially-observed discrete-event systems. *Automatica* (2009) 45:301–11. doi:10.1016/j.automatica.2008.06.022

2. Kumar R, Takai S. Decentralized prognosis of failures in discrete event systems. *IEEE Trans Autom Control* (2010) 55:48–59. doi:10.1109/TAC.2009.2034216

3. Takai S, Kumar R. Distributed failure prognosis of discrete event systems with bounded-delay communications. *IEEE Trans Autom Control* (2012) 57:1259–65. doi:10.1109/TAC.2011.2173419

4. Takai S, Kumar R. Distributed prognosis of discrete event systems under bounded-delay communications. In: Proc 48h IEEE Conf Decis Control, & 28th Chinese Control Conf; December 2009; Shanghai, China. IEEE (2009). p. 1235–40. doi:10.1109/CDC.2009.5399980

5. Chang M, Dong W, Ji Y, Tong L. On fault predictability in stochastic discrete event systems. *Asian J Control* (2013) 15:1458–67. doi:10.1002/asjc.748

6. Chen J, Kumar R. Stochastic failure prognosability of discrete event systems. *IEEE Trans Autom Control* (2015) 60:1570–81. doi:10.1109/TAC.2014.2381437

7. Liao H, Liu F, Wu N. Robust predictability of stochastic discrete-event systems and a polynomial-time verification. *Automatica* (2022) 144:110477. doi:10.1016/j.automatica.2022.110477

8. Benmessahel B, Touahria M, Nouioua F. Predictability of fuzzy discrete event systems. *Discrete Event Dyn Syst* (2017) 27:641–73. doi:10.1007/s10626-017-0256-7

9. Yin X, Li Z. Reliable decentralized fault prognosis of discrete-event systems. *IEEE Trans Syst Man Cybern: Syst* (2016) 46:1598–603. doi:10.1109/TSMC.2015.2499178

10. Takai S. Robust prognosability for a set of partially observed discrete event systems. *Automatica* (2015) 51:123–30. doi:10.1016/j.automatica.2014.10.104

11. Xiao C, Liu F. Robust fault prognosis of discrete-event systems against loss of observations. *IEEE Trans Autom Sci Eng* (2022) 19:1083–94. doi:10.1109/TASE.2021.3049400

12. Ammour R, Leclercq E, Sanlaville E, Lefebvre D. Fault prognosis of timed stochastic discrete event systems with bounded estimation error. *Automatica* (2017) 82:35–41. doi:10.1016/j.automatica.2017.04.028

13. Yin X. Verification of prognosability for labeled petri nets. *IEEE Trans Autom Control* (2018) 63:1828–34. doi:10.1109/TAC.2017.2756096

14. You D, Wang S, Seatzu C. Verification of fault-predictability in labeled petri nets using predictor graphs. *IEEE Trans Autom Control* (2019) 64:4353–60. doi:10.1109/TAC.2019.2897272

15. Sampath M, Sengupta R, Lafortune R, Sinnamohideen K, Teneketzis D. Diagnosability of discrete-event systems. *IEEE Trans Autom Control* (1995) 40:1555–75. doi:10.1109/9.412626

16. Alves MV, Barcelos RJ, Carvalho LK, Basilio JC. Robust decentralized diagnosability of networked discrete event systems against Dos and deception attacks. *Nonlinear Analysis: Hybrid Syst* (2022) 44:101162. doi:10.1016/j.nahs.2022.101162

17. Li Y, Hadjicostis CN, Wu N, Li Z. Error- and tamper-tolerant state estimation for discrete event systems under cost constraints. *IEEE Trans Autom Control* (2023) 1–8. doi:10.1109/TAC.2023.3239590

18. Zhang Q, Seatzu C, Li Z, Giua A. Joint state estimation under attack of discrete event systems. *IEEE Access* (2021) 9:168068–79. doi:10.1109/ACCESS.2021.3135870

19. Meira-Góes R, Kang E, Kwong RH, Lafortune S. Synthesis of sensor deception attacks at the supervisory layer of Cyber-Physical Systems. *Automatica* (2020) 121:109172. doi:10.1016/j.automatica.2020.109172