A Joint Diagnoser Approach for Diagnosability of Discrete Event Systems Under Attack

Tenglong Kang, Carla Seatzu, Zhiwu Li, and Alessandro Giua

February 2025

Abstract

This paper investigates the problem of diagnosing the occurrence of a fault event in a discrete event system (DES) subject to malicious attacks. We consider a DES monitored by an operator through the perceived sensor observations. It is assumed that an attacker can tamper with the sensor observations, and the system operator is not aware of the attacker's presence at the beginning. We propose a stealthy joint diagnoser (SJD) that (i) describes all possible stealthy attacks (i.e., undiscovered by the operator) in a given attack scenario; (ii) records the joint diagnosis state, i.e., the diagnosis state of the attacker consistent with the original observation and the diagnosis state of the operator consistent with the corrupted observation. The SJD is used for diagnosability verification under attack. From the attacker's point of view, we present two levels of stealthy attackers: one only temporarily degrades the diagnosis state of the operator, and the other permanently causes damage to the diagnosis state of the operator, thereby resulting in a violation of diagnosability. Finally, necessary and sufficient conditions for the existence of the two levels of attackers are presented.

Published as:
Tenglong Kang, Carla Seatzu, Zhiwu Li, and Alessandro Giua. "A Joint Diagnoser Approach for Diagnosability of Discrete Event Systems Under Attack", *Automatica*, vol. 172, p.112004, February, 2025.
DOI: 10.1016/j.automatica.2024.112004

The material in this paper was partially presented at the 62nd IEEE Conference on Decision and Control, December 13–15, 2023, Singapore. This work was partially supported by the National Key R&D Program of China under Grant 2018YFB1700104, the National Natural Science Foundation of China under Grant 61873342, the Science Technology Development Fund, MSAR under Grant No. 0029/2023/RIA1, project SERICS (PE00000014) under the MUR National Recovery, and Resilience Plan funded by the European Union - NextGenerationEU.

Tenglong Kang is with the School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China, and also with the Department of Electrical and Electronic Engineering, University of Cagliari, 09124 Cagliari, Italy. tlkang@stu.xidian.edu.cn.

Zhiwu Li (corresponding Author) is with the School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China, and also with the Institute of Systems Engineering, Macau University of Science and Technology, Macau, China. zhwli@xidian.edu.cn.

Carla Seatzu and Alessandro Giua are with the Department of Electrical and Electronic Engineering, University of Cagliari, 09124 Cagliari, Italy. {carla.seatzu, giua}@unica.it

1 Introduction

Fault diagnosis in discrete event systems (DESs) aims to determine whether some particular events, called faults, have occurred according to the current observation. Diagnosability is a property that guarantees the detection of any fault occurrence within a bounded delay. The verification can be done by using *diagnosers* [1, 2] or *verifier* [3]. It is well known that the diagnoser reported in [1, 2] has, in the worst case, exponential complexity in the size of the plant state space, but it can be used for both online diagnosis and offline diagnosability verification, unlike the verifier [3] which can only be used for diagnosability verification.

Over the past decade, there have been some works on *robust fault diagnosis* against sensor failures and communication failures; see, e.g., [4]. Such disruptions are not necessarily caused by malicious attacks with the goal of compromising the desired system properties. Indeed, the increasing number of networked components in systems possibly introduce vulnerabilities to cyber-attacks. In this paper, we consider a special class of cyber-attacks, called *sensor deception attacks*, that may obtain access to and compromise the transmitted signals between sensors attached to a system and an operator that monitors the system (see Fig. 2 for a sketch). Hence, the operator may be misled to take incorrect actions based on the corrupted information, which is our considered attack mechanism.

Recent works have explored sensor attacks in the context of attack detection [5], state estimation [6, 7], and supervisory control [8]. However, very few works have been devoted to diagnosability verification under attack. Two recent contributions in this context are [9] and [10]. In these works, from *defense viewpoint*, the notion of *robust diagnosability against attacks* is proposed, which requires that the system language can remain diagnosable even in the case of attacks.

In this paper, we develop a novel framework for diagnosis and diagnosability verification under attack. We take the perspective of the attacker. If an attacker may make a system no longer diagnosable, it is said to be *strongly harmful*. This occurs when the attacker is certain of the fault occurrence, while due to the attack, the operator is always not. In this context, we say that the attacker permanently conceals the fault occurrence to the operator. In the relaxed case, if the attacker might temporarily conceal the fault occurrence at some points after attacks occur, it is said to be *weakly harmful*. On the other hand, *stealthiness* is another crucial property of an attacker, ensuring that its attacks remain undiscovered by the operator during system execution. To our knowledge, this work is the first one that addresses the stealthiness of *active attacks against fault diagnosis* in the literature. To check stealthiness, we establish an attack detection mechanism by determining whether the operator perceives abnormal system observations.

From the *attack viewpoint*, we aim to verify the existence of attackers that can achieve harmfulness (including strong and weak harmfulness) and stealthiness. To this end, a bipartite diagnoser called *joint diagnoser* (*JD*) is constructed, which captures all possible attacks in a given attack scenario. We prove that the JD shows the joint diagnosis state for both the attacker (based on the original observation) and the operator (based on the corrupted observation). As a result, the JD provides a necessary and sufficient condition for the existence of a harmful attacker. In particular, we show that in our approach, diagnosability under attack can be verified by studying certain cycles in the JD, without the analysis of an indeterminate cycle as was the case for the classical diagnoser-based approach in [2].

Note that the JD also allows to determine if an attacker may actively make stealthy choices in its attack strategies. To capture stealthy attacks only, we present a refined JD, i.e., *stealthy joint diagnoser* (*SJD*) that is used to check the existence of a stealthy and harmful attacker. This is a distinguishing feature of our work as compared with prior works [9, 10]. In [9], the attacker's stealthiness is not considered. In [10], attack detection relies on the fault diagnosis technique by passively modeling an attack as a fault behavior under certain assumptions.

A preliminary version of this paper [11] introduced the diagnosis setting we adopt without providing proofs of the results. Here, we provide formal proofs and detailed examples. Further, we also present a novel approach for diagnosability verification under attack.

2 Preliminaries

In this paper, a plant is modeled as a *deterministic finite state automaton* (*automaton* for short) $G = (X, E, \delta, x_0)$, where X is a finite set of states, E is a finite set of events, $\delta : X \times E \to X$ is a partial transition function, and $x_0 \in X$ is an initial state. We use E^* to denote the *Kleene closure* of E, consisting of all words over E with finite lengths (including the empty word ε). Given a word $\sigma \in E^*$, $|\sigma|$ denotes the *length* of σ. The set of *prefixes* of a word σ is denoted as $\overline{\sigma} = \{u \in E^* \,|\, (\exists v \in E^*)\,[uv = \sigma]\}$. In G, a sequence $x_1 \xrightarrow{e_1} x_2 \xrightarrow{e_2} \cdots \xrightarrow{e_{l-1}} x_l (l \geq 2)$ of transitions such that $x_{h+1} = \delta(x_h, e_h)$ for all $h \in \{1, 2, \ldots, l-1\}$ is called a cycle if $x_1 = x_l$. A language $L \subseteq E^*$

is a subset of E^*. The *language generated by* G, denoted as $L(G)$ or, simply, L, is defined as $L(G) = \{\sigma \in E^* \mid \delta(x_0, \sigma) \text{ is defined}\}$. Given a word $\sigma \in L$, the *post-language of* L *after* σ is defined as $L/\sigma = \{t \in E^* \mid \sigma t \in L\}$. A language L is said to be live if for all $\sigma \in L$, there exists an event $e \in E$ such that $\sigma e \in L$.

Assuming that set E is divided into the *observable event set* E_o and the *unobservable event set* E_{uo}, the *natural projection* $P : E^* \to E_o^*$ is defined as

$$P(\varepsilon) = \varepsilon \text{ and } P(\sigma e) = \begin{cases} P(\sigma)e, & \text{if} \quad e \in E_o; \\ P(\sigma), & \text{if} \quad e \in E_{uo}. \end{cases}$$

The *inverse projection* $P^{-1} : E_o^* \to 2^{L(G)}$ is defined as $P^{-1}(s) = \{\sigma \in L(G) \mid P(\sigma) = s\}$, i.e., $P^{-1}(s)$ consists of all words σ in $L(G)$ whose observations are s. The *observed language* of G, denoted as $P(L(G))$ or, simply, $P(L)$, is defined as $P(L(G)) = \{s \in E_o^* \mid (\exists \sigma \in L(G))[s = P(\sigma)]\}$.

Let $G_1 = (X_1, E_1, \delta_1, x_{01})$ and $G_2 = (X_2, E_2, \delta_2, x_{02})$. The *parallel composition* of G_1 and G_2 is defined as $G_1 \parallel G_2 = Ac(X_1 \times X_2, E_1 \cup E_2, \delta, (x_{01}, x_{02}))$, where $\delta[(x_1, x_2), e] = (x_1', x_2')$ if $\delta_1(x_1, e) = x_1'$ and $\delta_2(x_2, e) = x_2'$; $\delta[(x_1, x_2), e] = (x_1', x_2)$ if $\delta_1(x_1, e) = x_1'$ and $e \notin E_2$; $\delta[(x_1, x_2), e] = (x_1, x_2')$ if $\delta_2(x_2, e) = x_2'$ and $e \notin E_1$; undefined, otherwise. In the definition of $G_1 \parallel G_2$, $Ac(\cdot)$ denotes the *accessible* (refer to [1]) part of an automaton.

2.1 Fault Diagnosis

Let $E_f \subseteq E_{uo}$ denote the set of fault events. In this paper, we consider a single class of fault for simplicity, but all the proposed results can be extended in a straightforward way to the case of multiple fault classes. Let $\Psi(E_f) = \{\sigma f \in L \mid \sigma \in E^*, f \in E_f\}$ denote the set of all finite words in L that end with a fault event f. With a slight abuse of notation, write $E_f \in \sigma$ to denote $\overline{\sigma} \cap \Psi(E_f) \neq \emptyset$. A word σ is said to be *faulty* (resp., *normal*) if $E_f \in \sigma$ (resp., $E_f \notin \sigma$). The language containing all normal words is denoted by $L_N \subset L$.

In the rest of this paper, the following usual assumptions hold: A1) The language of G is live; A2) There is no cycle of unobservable events in G. The fault diagnosis problem is to determine, based on the observation $s \in E_o^*$, if a fault has already occurred or not. To solve this problem, one wishes to build a *diagnosis function* $\gamma : E_o^* \to \{N, F, U\}$ associating to each observation a diagnosis state, such that

$$\gamma(s) = \begin{cases} N, & \text{if} \quad \forall \sigma \in P^{-1}(s), E_f \notin \sigma; \\ F, & \text{if} \quad \forall \sigma \in P^{-1}(s), E_f \in \sigma; \\ U, & \text{otherwise.} \end{cases}$$

In other words, an observation s is called: *normal* if $\gamma(s) = N$ since in this case no word producing s contains a fault; *faulty* if $\gamma(s) = F$ since in this case all words producing s contain a fault; *ambiguous* otherwise. A standard approach to compute the diagnosis function is by using a *diagnoser*. The diagnoser of G is defined as

$$Diag(G) = (X_d, E_o, \delta_d, x_{d,0}). \tag{1}$$

State $x_d \in X_d$ is in the form $x_d = \{(x_1, \ell_1), \cdots, (x_n, \ell_n)\}$, where $x_i \in X$ and $\ell_i \in \{N, F\}$ for $i = 1, \cdots, n$. The diagnoser allows one to associate every state to a diagnosis value $\gamma(x_d) = \gamma(s)$, where $x_d = \delta_d(x_{d,0}, s)$. The diagnoser state x_d is *negative* if $\gamma(x_d) = N$; *positive* if $\gamma(x_d) = F$; *uncertain* if $\gamma(x_d) = U$. Now we recall the notion of language diagnosability proposed in [2].

Definition 1: A language L is *diagnosable* w.r.t. $P : E^* \to E_o^*$ and E_f if

$$(\exists n \in \mathbb{N})(\forall \sigma \in \Psi(E_f))(\forall t \in L/\sigma)\,[|t| \geq n \Rightarrow \mathbf{C}_{\mathrm{D}}]$$

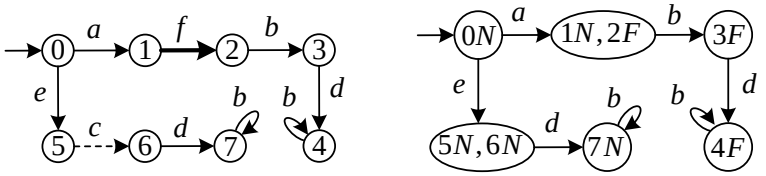
where $\mathbf{C}_{\mathrm{D}}$ is the diagnosability condition, defined as

$$(\nexists \sigma' \in L_N)\ [P(\sigma t) = P(\sigma')]\,,$$

with $\mathbb{N}$ denoting the set of non-negative integers. $\diamond$

In words, diagnosability guarantees that the occurrence of a fault event can be detected within a finite number of transitions after its occurrence. It was shown in [2] that a necessary and sufficient condition for diagnosability is the nonexistence of an *indeterminate cycle* in $Diag(G)$. An indeterminate cycle is a cycle composed exclusively of uncertain states in $Diag(G)$, which corresponds to two cycles in G, one that includes only states with label F and the other that includes only states with label N.

Example 1: Fig. 1(a) shows a plant G, where $E_o = \{a, b, d, e\}$, $E_{uo} = \{c, f\}$, and $E_f = \{f\}$. Since there are no cycles formed with uncertain states in the diagnoser shown in Fig. 1(b), which implies the absence of indeterminate cycles, we conclude that $L(G)$ is diagnosable. $\diamond$



(a) G (b) $Diag(G)$

Figure 1: (a) A plant G and (b) its diagnoser $Diag(G)$.

2.2 Sensor Attack

In this section, we consider an attacker that can compromise a subset of the sensor network channels [11, 8, 6, 7]. It may implement two types of sensor attacks:

- *Sensor Erasure attack* (*SE-attack*): erase some readings generated by the plant.
- *Sensor Insertion attack* (*SI-attack*): insert some fake readings that have not occurred in the plant.

Considering a system modeled as an automaton, we follow the notation in [6] and denote by $E_{era} \subseteq E_o$ (resp., $E_{ins} \subseteq E_o$) the set of events subject to SE-attacks (resp., SI-attacks), i.e., the occurrence of events in E_{era} (resp., E_{ins}) can be erased (resp., inserted). To make the problem more general, no relation is imposed between the sets E_{era} and E_{ins}. We define $E_{com} = E_{era} \cup E_{ins}$ as the *compromised event set*. Fig. 2 illustrates the architecture of a fault diagnosis system under attack, where the shadowed block denotes a sensor attacker that intervenes in the communication channels between the sensor and the system operator. If a plant generates a word $\sigma \in E^*$, the attacker observes an *original observation* $s = P(\sigma)$ and then produces a *corrupted observation*, denoted as s'. The operator monitors and diagnoses the system based on the received corrupted observation s', i.e., the diagnosis state $\gamma(s')$ is computed.

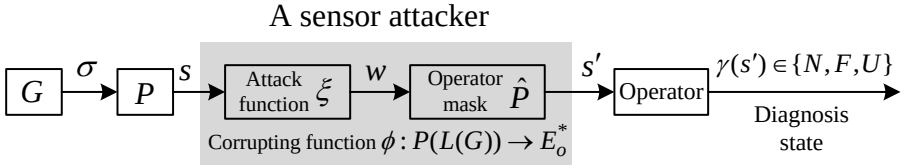


Figure 2: A fault diagnosis system under attack.

To identify the insertion or erasure of an event, we define $E_+ = \{e_+ \mid e \in E_{ins}\}$ as the set of *inserted events* and $E_- = \{e_- \mid e \in E_{era}\}$ as the set of *erased events*. The occurrence of event e_+ (resp., e_-) denotes the fact that the attacker inserts event $e \in E_{ins}$ that has not occurred in the plant (resp., event $e \in E_{era}$ has been erased by the attacker). Note that $E_o, E_+,$ and E_- are disjoint sets. We also define $E_a = E_o \cup E_+ \cup E_-$ as the *attack alphabet*.

Definition 2: Given a plant G with set $E_{com} = E_{era} \cup E_{ins}$, an *attack function* $\xi : P(L(G)) \to E_a^*$ satisfies:
1) $\xi(\varepsilon) \in E_+^*$;
2) $\forall se \in P(L(G))$ with $s \in E_o^*$ and $e \in E_o$:

$$\begin{cases} \xi(se) \in \xi(s)\{e\}E_+^*, & \text{if } e \in E_o \backslash E_{era}; \\ \xi(se) \in \xi(s)\{e_-, e\}E_+^*, & \text{if } e \in E_{era}. \end{cases} \qquad \diamond$$

Statement 1) in Definition 2 implies that an arbitrary word $t \in E_+^*$ may be initially inserted. By Statement 2), the attacker cannot erase e if e is outside of E_{era}, but can insert an arbitrary word $t \in E_+^*$ after e. If an event $e \in E_{era}$ occurs, the attacker may either erase e or leave it intact, and then insert any arbitrary word $t \in E_+^*$. Note that the attack function ξ is deterministic. We define the set of all possible attack functions depending on sets E_{era} and E_{ins} as "$\Xi(E_{era}, E_{ins})$" (abbreviated as Ξ).

An attack function ξ determines an *attack language*, denoted as $L(\xi, G) = \{\xi(s) \mid s \in P(L(G))\}$ (abbreviated as $L(\xi)$). Let $w \in L(\xi)$ be an *attack word*. Given an attack word w, to capture the corrupted observation s' perceived by the operator, an *operator mask* $\widehat{P} : E_a^* \to E_o^*$ is defined as

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

The operator mask can be extended to the language $L(\xi)$ by applying $\widehat{P}(w)$ to all words $w \in L(\xi)$. We now define two diagnosis functions on E_a as follows. The *attacker diagnosis function* $\gamma_{att} : E_a^* \to \{N, F, U\}$ is defined as $\gamma_{att}(w) = \gamma(s)$, where $w = \xi(s)$; the *operator diagnosis function* $\gamma_{opr} : E_a^* \to \{N, F, U\}$ is defined as $\gamma_{opr}(w) = \gamma(s')$, where $s' = \widehat{P}(w)$.

Definition 3: A *corrupting function* $\phi : P(L(G)) \to E_o^*$ is defined as $\phi(s) = \widehat{P}(\xi(s))$, where $\xi : P(L(G)) \to E_a^*$ and $\widehat{P} : E_a^* \to E_o^*$ are the attack function and the operator mask, respectively. $\diamond$

In this paper, a sensor attacker is characterized by a corrupting function that takes an original observation s as input and produces a corrupted observation $s' = \phi(s) = \widehat{P}(\xi(s))$ as output, as shown in Fig. 2. We define $L(\phi, G) = \widehat{P}(L(\xi, G))$, abbreviated as $L(\phi)$, as the *corrupted language* induced by function ϕ. We also define the set of all possible corrupting functions that depend on E_{era} and E_{ins} as "$\Phi(E_{era}, E_{ins})$" abbreviated as Φ.

3 Active attacks on fault diagnosis

We start this section with a motivating example to show that, in the case of sensor attacks, the attacker may actively degrade the estimation performance.

Example 2: Let us continue with Example 1. In the nominal setting, if word $\sigma = afbdb^k$ ($k \in \mathbb{N}$) occurs, then the diagnoser $Diag(G)$ goes from the initial state $\{0N\}$ to $\{4F\}$. Now assume that the sensors that record the occurrence of a and b are subject to SE-attacks, and the sensor that records the occurrence of e is subject to SI-attacks, i.e., $E_{era} = \{a, b\}$ and $E_{ins} = \{e\}$. We consider an attacker ϕ with the following attack function

$$\xi(s) = \begin{cases} \varepsilon, & \text{for} \quad s = \varepsilon; \\ a\_b\_e_+db^k, & \text{for} \quad s = abdb^k, k \in \mathbb{N}. \end{cases}$$

In words, the attacker first erases event a and b, and then inserts event e while observing $abdb^k$. Under such an attack, the first event occurrence to be recognized by $Diag(G)$ is the inserted event e, which takes the diagnoser state to $\{5N, 6N\}$. When the next event of σ, i.e., event d occurs, the diagnoser moves to state $\{7N\}$, where it stays as long as event b continues to occur, therefore displaying wrong information regarding the fault occurrence. This false negative shows that the diagnosis state based on the original observation $s = abdb^k$ may not coincide with the diagnosis state based on the corrupted observation $s' = \widehat{P}(\xi(abdb^k)) = edb^k$. $\diamond$

3.1 Problem statement

Before formalizing the above discussion, in addition to A1 and A2, we make the following assumption: A3) The language of G is diagnosable in the nominal setting, i.e., when no attacks occur. Example 2 motivates the notion of *weakly harmful attackers* (*WH-attackers*).

Definition 4: An attacker ϕ for a language L is said to be *weakly harmful* if there exists an observation $s \in P(L)$ with $\gamma(s) = \{F\}$ which may be corrupted into another observation $s' = \phi(s)$ and $\gamma(s') \in \{N, U\}$. $\diamond$

As defined, a WH-attacker can degrade the diagnoser performance in such a way that a faulty observation s that allows the detection of a fault is altered into a normal or ambiguous observation s' received by the operator, which corresponds to the absence of the fault or to the uncertain situation, respectively. Such an attacker introduces a delay in the detection of the fault, i.e., it makes a fault temporarily hidden from the operator. Now we formalize the *WH-attacker Existence Problem*.

Problem 1: Given a language L, determine whether there exists a WH-attacker for L. $\diamond$

Further, a WH-attacker may lead to a violation of diagnosability. We now give a language diagnosability condition that takes into account a class of sensor attackers depending on E_{era} and E_{ins}.

Definition 5: A language L is *robustly diagnosable against* $\Phi(\Sigma_{era}, \Sigma_{ins})$ w.r.t. $P : E^* \to E_o^*$ and E_f if

$$(\exists n \in \mathbb{N})(\forall \sigma \in \Psi(E_f))(\forall t \in L/\sigma)\,[|t| \geq n \Rightarrow C_{AD}]$$

where condition C_{AD} is given as

$$(\forall \phi \in \Phi(E_{era}, E_{ins}))(\nexists \sigma' \in L_N)\,[\phi(P(\sigma t)) = P(\sigma')] \quad \diamond$$

This definition can be expressed as follows. Assume that the plant generates a word $\sigma \in \Psi(E_f)$ and the evolution continues. After a finite number of steps n (that depends on σ), any attacker $\phi \in \Phi(E_{ins}, E_{era})$ observes $P(\sigma t)$ with $|t| \geq n$ and produces the corrupted observation $\phi(P(\sigma t))$, which is a faulty observation consistent with no normal word $\sigma' \in L_N$. With the above concepts, we introduce the following fault diagnosis problem.

Problem 2: (Diagnosability Under Attack) Given a language L, determine whether L is robustly diagnosable against attacks. $\diamond$

This problem can be investigated by exploring the existence of *strongly harmful attackers* (*SH-attackers*). The negation of Definition 5 involves the formal notion of SH-attackers as follows.

Definition 6: A sensor attacker $\phi \in \Phi(E_{era}, E_{ins})$ for a language L is said to be *strongly harmful* if

$$(\forall n \in \mathbb{N})(\exists \sigma \in \Psi(E_f))(\exists t \in L/\sigma)\,[|t| \geq n \wedge C_{\text{NAD}}]$$

where condition C_{NAD} is defined as

$$(\exists \sigma' \in L_N)\ [\phi(P(\sigma t)) = P(\sigma')]\,. \qquad \diamond$$

This definition implies that there exists a faulty word σ which can be extended with an arbitrarily long word t in such a way that all observations $P(\sigma t)$ are corrupted by the attacker into a normal or ambiguous observation $\phi(P(\sigma t))$, which is identical to the observation produced by a normal word σ'. The SH-attacker introduces an infinite delay in the detection of the fault, i.e., permanently conceals the fault occurrence to the operator, thereby resulting in a violation of diagnosability. Hence, Problem 2 is equivalent to the *SH-attacker Existence Problem*.

Example 3: Consider the attacker described in Example 2. We focus on the following words in $L(G)$: $\sigma t = afbdb^k$, and $\sigma' = ecdb^k$. For the faulty word $afbdb^k$, the attacker ϕ observes $s = abdb^k$ and produces $s' = \phi(abdb^k) = \widehat{P}(\xi(abdb^k)) = \widehat{P}(a\_b\_e_+db^k) = edb^k$, which is a normal observation consistent with $\sigma' = ecdb^k$. This implies that condition C_{NAD} in Definition 6 holds. Hence, ϕ is an SH-attacker. $\diamond$

3.2 Attacker's stealthiness

Due to the effect of sensor attacks on the system diagnosability, another issue of interest is attack detection, in particular, the detection of attacks that violate diagnosability. To this end, we introduce an attack detection mechanism [8, 6]. It is assumed that the attacker and the operator have full knowledge of the plant, while the operator does not realize the attacker's presence at the beginning. If an attacker can always keep its attacks undiscovered by the operator during system execution, it is said to be *stealthy*, defined as follows.

Definition 7: A sensor attacker ϕ for a language L is said to be *stealthy* if $L(\phi) = \widehat{P}(L(\xi)) \subseteq P(L(G))$. $\diamond$

The stealthiness of an attacker is guaranteed when any corrupted observation perceived by the operator is contained in the observed language of G. Note that Definition 7 requires the attacker not only to be undiscovered at the point when attacks occur but also to remain stealthy no matter how the system evolves in the future. Consider the stealthiness of the attacker ϕ in Example 3. Since $L(\phi) \subseteq P(L(G))$, we conclude that ϕ is stealthy.

We now define two sets of words $w \in E_a^*$ as follows. Given a plant G, the set of *stealthy words* on E_a is defined as $\mathcal{W}_s = \{w \in E_a^* \mid \widehat{P}(w) \in P(L(G))\}$, while the set of *exposing words* on E_a is defined as $\mathcal{W}_e = \{we_a \in E_a^* \mid w \in \mathcal{W}_s, e_a \in E_a, we_a \notin \mathcal{W}_s\}$. A stealthy word w produces a corrupted observation $s' = \widehat{P}(w) \in P(L(G))$, which does not reveal the attacker's presence. On the contrary, an exposing word results in the exposure of the attacker at the last step.

From an attacker's viewpoint, the goals of affecting fault diagnosis and keeping stealthy are separated. In this regard, attackers that can achieve both goals are: *stealthy weakly harmful attackers* (*SWH-attackers*) and *stealthy strongly harmful attackers* (*SSH-attackers*).

Problem 3: Given a language L, determine whether there exists an SWH-attacker for L. ◇

Problem 4: Given a language L, determine whether there exists an SSH-attacker for L. ◇

4 Stealthy joint diagnoser

This section introduces an information structure called *joint diagnoser* which describes for all possible attack words, the diagnosis state corresponding to the original observation, and the diagnosis state corresponding to the corrupted observation.

4.1 Attacker diagnoser and operator diagnoser

To construct a joint diagnoser, we first briefly review the constructions of two augmented diagnosers: *Attacker Diagnoser* and *Operator Diagnoser*, proposed in [11]. The former describes for all attack words w that can be generated under attack which is the diagnosis state computed by the attacker. The latter describes for all words $w \in \mathcal{W}_s \cup \mathcal{W}_e$ which is the diagnosis state computed by the operator. From a nominal diagnoser $Diag(G) = (X_d, E_o, \delta_d, x_{d,0})$, we build:

- *Attacker Diagnoser* $Diag_{att}(G) = (X_d, E_a, \delta_{att}, x_{d,0})$ by self-looping each state with all events in E_+, and then adding in parallel to each event $e \in E_{era}$ the corresponding event $e_- \in E_-$.

- *Operator Diagnoser* $Diag_{opr}(G) = (X_d \cup d_\emptyset, E_a, \delta_{opr}, x_{d,0})$ by self-looping each state with all events in E_-, then adding in parallel to each event $e \in E_{ins}$ the corresponding event $e_+ \in E_+$, and finally adding a dump state $d_\emptyset$ that is reached by all undefined transitions.

Example 4: Consider the plant G in Fig. 1(a). The corresponding $Diag_{att}(G)$ and $Diag_{opr}(G)$ are shown in Fig. 3(a) and (b), respectively. At each state of $Diag_{att}(G)$, all transitions labeled with the inserted event e_+ are in self-loop, since the attacker knows that the inserted event is fake. There exists a transition labeled with the erased event a_- from state $\{0N\}$ to $\{1N, 2F\}$ in $Diag_{att}(G)$ since the attacker knows that the event a has occurred. At each state of $Diag_{opr}(G)$, all transitions labeled with the erased events a_- and b_- are in self-loop, since the operator cannot perceive their occurrences after erasure. There exists a transition labeled with the inserted event e_+ from state $\{0N\}$ to $\{5N, 6N\}$ in $Diag_{opr}(G)$ since the operator cannot distinguish between e_+ and the corresponding event e. ◇

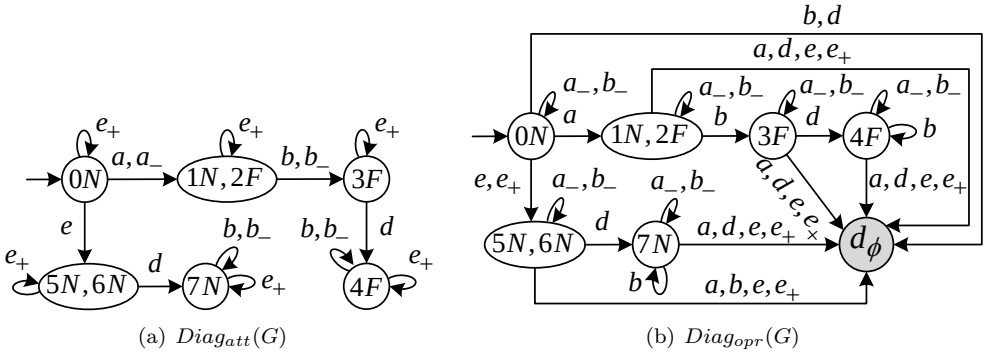


Figure 3: (a) $Diag_{att}(G)$ and (b) $Diag_{opr}(G)$ for G in Fig. 1(a).

Proposition 1: Given a nominal diagnoser $Diag(G) = (X_d, E_o, \delta_d, x_{d,0})$, let $Diag_{att}(G) = (X_d, E_a, \delta_{att}, x_{d,0})$ be the attacker diagnoser. It holds that

1) $w \in L(Diag_{att}(G))$

$\Leftrightarrow (\exists \xi \in \Xi)(\exists s \in L(Diag(G))\,[w = \xi(s)]$;

2) $(\forall \xi \in \Xi)(\forall s \in L(Diag(G)))[\delta_{att}(x_{d,0}, \xi(s)) = \delta_d(x_{d,0}, s)]$

Proof: The proof is carried out by recursively considering all possible attack functions. First, we focus on the "null attack function" that produces no insertions or erasures. Clearly, given a word $w \in E_o^*$, the predicate $w \in L(Diag(G)) \Leftrightarrow w \in L(Diag_{att}(G))$ holds with $\delta_{att}(x_{d,0}, w) = \delta_d(x_{d,0}, w)$. Now suppose that the statements hold for a given attack function ξ. Consider an attack function ξ' that contains just one more insertion action than ξ after some observation s_1. This means that there exists $s_1 \in L(Diag(G))$ with $\xi(s_1) = w_1$ and $e \in E_{ins}$ such that for all $s = s_1 s_2 \in L(Diag(G))$, it holds that $\xi(s) = w_1 w_2$ and $\xi'(s) = w_1 e_+ w_2$, while for all other words $s \in L(Diag(G))$, we have $\xi(s) = \xi'(s)$. Hence, it comes that $\xi'(s) \in L(Diag_{att}(G))$ and $\delta_{att}(x_{d,0}, \xi'(s)) = \delta_{att}(x_{d,0}, \xi(s)) = \delta_d(x_{d,0}, s)$, since all events $e_+ \in E_+$ are self-looped at each state of $Diag_{att}(G)$.

Consider an attack function ξ' that contains just one more erasure action than ξ. This means that there exists $s_1 \in L(Diag(G))$ with $\xi(s_1) = w_1$ and $e \in E_{era}$ such that for all $s = s_1 e s_2 \in L(Diag(G))$, it holds that $\xi(s) = w_1 e w_2$ and $\xi'(s) = w_1 e_- w_2$, while for all other words $s \in L(Diag(G))$, we have $\xi(s) = \xi'(s)$. Clearly, it comes that $\xi'(s) \in L(Diag_{opr}(G))$ and $\delta_{opr}(x_{d,0}, \xi'(s)) = \delta_{opr}(x_{d,0}, \xi(s)) = \delta_d(x_{d,0}, s)$, since for all events $e \in E_{era}$ in $Diag_{opr}(G)$, there exists a parallel event $e \in E_-$. As this inductive procedure generates the set Ξ of all attack functions, the result follows. □

By Statement 1) in Proposition 1, the language of the attacker diagnoser consists of all possible attack words w, which may correspond to an original observation s produced by G (also perceived by the attacker). According to Statement 2), the diagnosis state estimation of $Diag_{att}(G)$ based on w coincides with that of $Diag(G)$ based on s, where $w = \xi(s)$.

Proposition 2: Given a nominal diagnoser $Diag(G) = (X_d, E_o, \delta_d, x_{d,0})$, let $Diag_{opr}(G) = (X_d \cup d_\emptyset, E_a, \delta_{opr}, x_{d,0})$ be the operator diagnoser. It holds that

1) $L(Diag_{opr}(G)) = \mathcal{W}_s \cup \mathcal{W}_e$;

2) $w \in \mathcal{W}_s \subseteq L(Diag_{opr}(G))$

$\Rightarrow (\exists s' \in L(Diag(G)))[s' = \widehat{P}(w)]$;

3) $\forall w \in L(Diag_{opr}(G))$: if $w \in \mathcal{W}_s$, $\delta_{opr}(x_{d,0}, w) = \delta_d(x_{d,0}, \widehat{P}(w))$; if $w \in \mathcal{W}_e$, $\delta_{opr}(x_{d,0}, w) = d_\emptyset$.

Proof: Statement 1). By construction, $Diag_{opr}(G)$ includes all words in $\mathcal{W}_s \cup \mathcal{W}_e$. We need to prove that all the words $w \in L(Diag_{opr}(G))$ either belong to $\mathcal{W}_s$ or $\mathcal{W}_e$. Consider a word $w \in L(Diag_{opr}(G))$ that reaches a state $x_d \in X_d$ and only contains the events in E_o, implying that no attack has been performed. At each state all events e_- are in self-loop, which corresponds to the generation of w. By the definition of $\widehat{P}$, it holds that $\widehat{P}(w) \in P(L(G))$, i.e., $w \in \mathcal{W}_s$. If the word w is generated by executing a transition $\delta_{opr}(x'_d, e_+) = x''_d$ with $e \in E_{ins}$, it is also possible to execute the "parallel" transition $\delta_{att}(x'_d, e) = x''_d$ and thus $\widehat{P}(w) \in P(L(G))$, i.e., $w \in \mathcal{W}_s$. Then, if the word w yields the dump state $d_\emptyset$, then $\widehat{P}(w) \notin P(L(G))$, i.e., $w \in \mathcal{W}_e$, which completes the proof of Statement 1). Similar to Proposition 1, Statements 2) and 3) can be proved by induction. □

By Statement 1) in Proposition 2, all words in $\mathcal{W}_s$ and $\mathcal{W}_e$ can be generated by $Diag_{opr}(G)$. By Statement 2), a word in $\mathcal{W}_s$ generated by $Diag_{opr}(G)$ is perceived by the operator as a corrupted observation $s' \in P(L(G))$. Statement 3) implies that: (i) the diagnosis state estimation of $Diag_{opr}(G)$ based on a stealthy word $w \in \mathcal{W}_s$ coincides with that of $Diag(G)$ based on s', where $s' = \widehat{P}(w)$; (ii) all exposing words $w \in \mathcal{W}_e$ yield $d_\emptyset$.

4.2 Joint diagnoser and refining process

Definition 8: A *joint diagnoser* (*JD* for short) $J\text{-}Diag(G)$ is defined as $J\text{-}Diag(G) = (R, E_a, \delta_a, r_0) = Diag_{att}(G) \| Diag_{opr}(G)$. ◇

As defined, each state of $J\text{-}Diag(G)$ is a pair $r = (x_d, \bar{x}_d)$. We define the set of exposing states as $R_e = \{(x_d, \bar{x}_d) \in R \,|\, \bar{x}_d = d_\emptyset\}$ reached by exposing words (those in $\mathcal{W}_e$), and the set of stealthy states as $R_s = R \backslash R_e$ reached by stealthy words (those in $\mathcal{W}_s$). An attacker that aims to remain stealthy should never produce an attack word $w \in \mathcal{W}_e$ yielding an exposing state in R_e in $J\text{-}Diag(G)$. However, there may exist stealthy states in R_s from which, following some future evolution of the plant G, an exposing state will necessarily be reached regardless of all future attacker's attempts (including insertions and erasures) to remain stealthy. This leads to the notion of *weakly exposing region*, denoted as $R_{we} \supseteq R_e$, which can be computed iteratively by a procedure in [7]. In the first iteration,

$$R_{we} := \{r \in R \mid (\exists e \in E_o)[\delta_a(r, e) \in R_e \Rightarrow \\ e \notin E_{era} \wedge (\forall e' \in E_{ins})[\delta_a(r, e') \in R_e]]\}. \quad (2)$$

The remaining iterations are executed similar to Eq. (2). We do not present the complete procedure here for the sake of brevity but illustrate it via Example 5. Dually, we define the *strongly stealthy region* as $R_{ss} = R \backslash R_{we} \subseteq R_s$.

Example 5: Continue Example 4. The joint diagnoser for G is shown in Fig. 4, where the exposing states in R_{we} are highlighted in gray, while the stealthy states in R_{we} are highlighted in brown. For instance, a stealthy state $(\{3F\}, \{1N, 2F\})$ is in R_{we}, since by Eq. (2) from this state, there exists an unerased event $d \notin E_{era}$ yielding an exposing state $(\{4F\}, \{x_{d,\emptyset}\})$, and meanwhile the inserted event $e \in E_{ins}$ also reaches an exposing state $(\{3F\}, \{x_{d,\emptyset}\})$. In plain words, once such a stealthy state is reached, all attempts of an attacker to prevent it from reaching a subsequent exposing state will fail. ◇

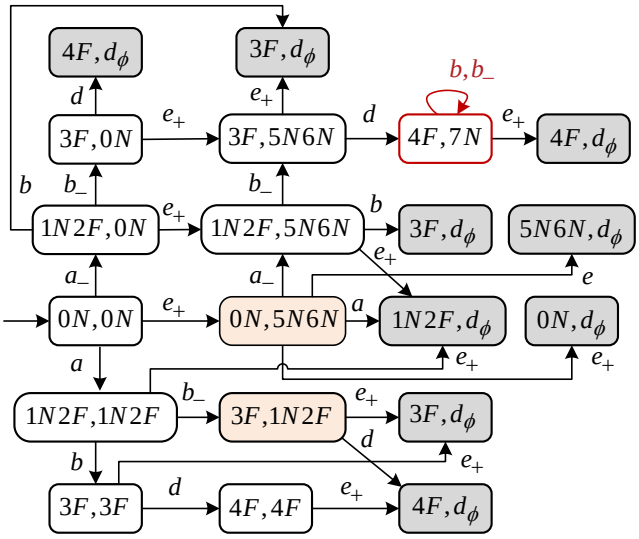


Figure 4: $J\text{-}Diag(G)$ in Example 5.

To characterize the set of all possible stealthy attackers, we can refine J-Diag(G) to ensure that no attack word reaching a state in R_{we} is produced.

Definition 9: Let $J\text{-}Diag(G) = (R, E_a, \delta_a, r_0)$ be a joint diagnoser. The *stealthy joint diagnoser* (*SJD*, for short) is defined as $SJ\text{-}Diag(G) = Ac(R_{ss}, E_a, \delta_{sa}, r_0)$, where $R_{ss} = R \backslash R_{we}$ and $\delta_{sa} = \delta_a|_{R_{ss} \times E_a \to R_{ss}}$. ◇

The resulting SJD is obtained by removing all states in R_{we} from a JD and taking its accessible part.

Theorem 1: Given a plant G, let $Diag(G) = (X_d, E_o, \delta_d,$
$x_{d,0})$ be the nominal diagnoser and $SJ\text{-}Diag(G) = (R_{ss}, E_a, \delta_{sa}, r_0)$ be the SJD. It holds that:

$$(\forall \xi \in \Xi)(\forall s \in L(Diag(G))) \\ [\delta_{sa}(r_0, \xi(s)) = (x_d, \bar{x}_d) \Leftrightarrow x_d = \delta_d(x_{d,0}, s) \wedge \\ \bar{x}_d = \delta_d(x_{d,0}, \widehat{P}(\xi(s))) \neq d_\emptyset]$$

Proof: ($\Leftarrow$) Assume that $\delta_d(x_{d,0}, s) = x_d$ and $\delta_d(x_{d,0}, \widehat{P}(\xi(s))) = \bar{x}_d$. By Propositions 1 and 2, and by $\bar{x}_d \neq d_\emptyset$, it holds that $\delta_d(x_{d,0}, s) = \delta_{att}(x_{d,0}, \xi(s))$ and $\delta_d(x_{d,0}, \widehat{P}(\xi(s))) = \delta_{opr}(x_{d,0}, \xi(s))$, i.e., $\delta_{att}(x_{d,0}, \xi(s)) = x_d$ and $\delta_{opr}(x_{d,0}, \xi(s)) = \bar{x}_d$. By $J\text{-}Diag(G) = Diag_{att}(G) \parallel Diag_{opr}(G)$, and by the construction of $SJ\text{-}Diag(G)$, it is $\delta_{sa}(r_0, \xi(s)) = (x_d, \bar{x}_d)$. ($\Rightarrow$) Assume that $\delta_{sa}(r_0, \xi(s)) = (x_d, \bar{x}_d)$. By $J\text{-}Diag(G) = Diag_{att}(G) \parallel Diag_{opr}(G)$, and by the construction of $SJ\text{-}Diag(G)$, it holds that $\delta_{att}(x_{d,0}, \xi(s)) = x_d$ and $\delta_{opr}(x_{d,0}, \xi(s)) = \bar{x}_d \neq d_\emptyset$. By Propositions 1 and 2, it is $\delta_d(x_{d,0}, s) = \delta_{att}(x_{d,0}, \xi(s))$ and $\delta_d(x_{d,0}, \widehat{P}(\xi(s))) = \delta_{opr}(x_{d,0}, \xi(s))$, i.e., $\delta_d(x_{d,0}, s) = x_d$ and $\delta_d(x_{d,0}, \widehat{P}(\xi(s))) = \bar{x}_d$. □

Hence, a state pair $r_{ss} = (x_d, \bar{x}_d)$ in the SJD reached by an attack word $w = \xi(s)$ describes the *joint diagnosis state estimate*, where x_d represents the *correct diagnosis state estimate* of the attacker for the original observation s, and $\bar{x}_d$ represents the *corrupted diagnosis state estimate* of the operator based on the corrupted observation $s' = \widehat{P}(w)$.

Finally, we conclude this section by the complexity analysis of the proposed approach. Let $Diag(G)$ be a nominal diagnoser with $|X_d|$ states. By construction, the attacker diagnoser $Diag_{att}(G)$ shares the same set of states of $Diag(G)$; so does the operator diagnoser $Diag_{opr}(G)$ except for the dump state $d_\emptyset$. Since the JD $J\text{-}Diag(G)$ is the parallel composition of $Diag_{att}(G)$ and $Diag_{att}(G)$, its maximum number of states is $|X_d| \times |X_d + 1|$. Moreover, determining if a state of $J\text{-}Diag(G)$ is in the strongly stealthy region has linear complexity in the size of $J\text{-}Diag(G)$. Hence, the complexity of building an SJD is $O(|X_d|^2)$, which is polynomial in the number of states of the nominal diagnoser. However, it is well known that the construction of the diagnoser is worst-case exponential with respect to the number of states in the system. As a result, the overall computational complexity of an SJD is exponential with respect to the number of states of G.

5 Diagnosability analysis under attack

We show that the proposed JD (resp., the SJD) leads to the solution of Problems 1 and 2 (resp., Problems 3 and 4). The following definition is first required.

Definition 10: Given an attack word w, a *diagnosis pair function* $d : E_a^* \to \{N, F, U\} \times \{N, F, U\}$ associating to $w \in E_a^*$ a diagnosis state pair is defined as $d(w) = (\gamma_{att}(w), \gamma_{opr}(w))$, where γ_{att} and γ_{opr} are the attacker and operator diagnosis functions, respectively. ◇

From the definitions of γ_{att} and γ_{opr}, it holds that $d(w) = (\gamma_{att}(w), \gamma_{opr}(w)) = (\gamma(s), \gamma(s'))$, where $w = \xi(s)$ and $s' = \widehat{P}(w)$. For any ξ, the diagnosis pair function can be computed by using the SJD. Let $r_{ss} = (x_d, \bar{x}_d) = \delta_{sa}(r_0, w)$. By Theorem 1, the SJD allows one to associate every state to a diagnosis state pair $d(r_{ss}) = d(w)$, i.e., $\gamma(x_d) = \gamma(s)$ and $\gamma(\bar{x}_d) = \gamma(s')$ are the diagnosis state of the attacker and operator, respectively.

The set of all diagnosis state pairs is defined as $D = \{N, F, U\} \times \{N, F, U\}$, which can be partitioned into $D = D_c \cup D_w \cup D_h$, where $D_c = \{(N, N), (U, U), (F, F)\}$, $D_w = \{(N, U), (N, F), (U, N), (U, F)\}$, and $D_h = \{(F, N), (F, U)\}$. The motivation for this partition will be clear later.

Definition 11: Let $SJ\text{-}Diag(G)$ be an SJD. A state r_{ss} is *correct* if $d(r_{ss}) \in D_c$; *wrong non-harmful* if $d(r_{ss}) \in D_w$; *harmful* if $d(r_{ss}) \in D_h$. Denote the set of correct states, the set of wrong non-harmful states, and the set of harmful states by R_{sc}, R_{sw}, and R_{sh}, respectively. ◇

When the SJD is in a correct state, the operator correctly computes the diagnosis state regardless of the fact that an attack has occurred. When the SJD reaches a wrong non-harmful state, the operator computes a wrong diagnosis state based on the corrupted observation due to an attack, which is inconsistent with the diagnosis state based on the original observation. Note that, in such a case, the fault diagnosis is manipulated due to the attack but does not pose a real danger.

Finally, harmful states of the SJD correspond to the detection of the fault based on the original observation, and no detection based on corrupted observation. Its physical interpretation is that the attacker itself has already confirmed that the fault has occurred, but it induces the operator to be unable to claim the fault occurrence. Intuitively, the presence of a harmful state is related to Problem 3.

Theorem 2: Given a plant G, there exists an SWH-attacker if and only if the SJD $SJ\text{-}Diag(G)$ contains a harmful state, i.e., $R_{sh} \neq \emptyset$.

Proof: (If) Assume that there exists a harmful state r_{ss} in $SJ\text{-}Diag(G)$ such that $d(r_{ss}) \in D_h$, where $r_{ss} = \delta_{sa}(r_0, w)$. By Theorem 1, associated with r_{ss} is an attack word w such that $d(w) = d(r_{ss}) \in D_h$. Hence, there exists a sensor attacker ϕ that alters the observation s into s' such that $(\gamma(s), \gamma(s')) = (\gamma_{att}(w), \gamma_{opr}(w)) = d(w)$, i.e., $(\gamma(s), \gamma(s')) \in \{(F,N), (F,U)\}$. By Definition 4, the sensor attacker ϕ is weakly harmful. By the construction of $SJ\text{-}Diag(G)$, ϕ is also stealthy.

(Only if) Assume that there exists a SWH-attacker ϕ. By Definition 4, there exists an observation s such that s is corrupted into s' satisfying $(\gamma(s), \gamma(s')) \in \{(F,N), (F,U)\}$. By Theorem 1, there necessarily exists a state r_{ss} in $SJ\text{-}Diag(G)$ by executing word w such that $d(r_{ss}) = (\gamma(x_d), \gamma(\bar{x}_d)) = (\gamma(s), \gamma(s')) \in \{(F,N), (F,U)\}$, i.e., $d(r_{ss}) \in D_h$. Hence, $R_{sh} \neq \emptyset$. □

In many practical cases, the attacker may only be interested in the impact of fault diagnosis, while it is willing to accept the risk of being discovered. Based on this consideration, we give the following result to solve Problem 1 (that is a relaxation of Problem 3).

Corollary 1: Given a plant G, there exists a WH-attacker if and only if the JD $J\text{-}Diag(G)$ contains a harmful state.

This corollary can be considered as a relaxation of Theorem 2; its proof is similar to that of Theorem 2. An example is provided to illustrate the above results.

Example 6: The corresponding SJD is omitted since it is part of the JD in Fig. 4 without states in R_{we}. Let $w_1 = a_- e_+$ be an attack word that yields the wrong non-harmful state $(\{1N, 2F\}, \{5N, 6N\})$. This implies that the diagnosis state of the attacker based on $s = a$ is "U", while the diagnosis state of the operator based on $s' = e$ is "N". At this point, the attacker has doubted if the fault has occurred or not; however, the operator is certain that the fault has not occurred.

Let the evolution continue. Another word $w_2 = a_- e_+ b_- d$ yields the harmful state $(\{4F\}, \{7N\})$. At this point, the attacker is certain that the fault has occurred based on $s = abd$; however, the operator persists in its opinion that the fault has not occurred based on the corrupted observation $s' = ed$. By Theorem 2, there exists an SWH-attacker. It produces the attack word $a_- e_+ b_- d$ by first erasing the occurrence of event a, then inserting e_+, and finally erasing event b while observing abd.

By contrast, let us focus on $w = ab_-$ that yields another harmful state $(\{3F\}, \{1N, 2F\})$. By Corollary 1, there exists a WH-attacker that produces the attack word $w = ab_-$. However, the attack word $w = ab_-$ yields a state $\{3F, 1N2F\} \in R_{we}$ as discussed in Example 5. Therefore, the WH-attacker is not stealthy. This conclusion can also be reached in terms of language by $\phi(abd) = ad \notin P(L(G))$. ◇

As discussed before, when the SJD of G reaches a harmful state, the attacker currently conceals to the operator the fact that a fault has occurred in G. If it further remains indefinitely in a cycle formed with harmful states, it is possible that the attacker may permanently conceal the fault occurrence to the operator. In addition, the cycle should not be exclusively caused by the inserted events. The reason is that by Definition 6, the existence of an SH-attacker implies a sufficiently long faulty word, but along the cycle exclusively formed with the inserted events, the number of events generated after the fault does not increase. Motivated by this, we provide the following result to solve Problem 4.

Theorem 3: Given a plant G, there exists an SSH-attacker if and only if there exists a reachable cycle

$$cl = r_{ss,1} \xrightarrow{e_1} r_{ss,2} \xrightarrow{e_2} \cdots \xrightarrow{e_{l-1}} r_{ss,l} \xrightarrow{e_1} r_{ss,1}$$

in the SJD $SJ\text{-}Diag(G)$, satisfying condition:

$$(\exists\, j \in \{1, 2, \ldots, l\})\ [r_{ss,j} \in R_{sh} \wedge e_j \in E_o \cup E_-]\,. \tag{3}$$

Proof: (If) Assume that in $SJ\text{-}Diag(G)$ there exists a cycle $r_{ss,1} \xrightarrow{e_1} r_{ss,2} \xrightarrow{e_2} \cdots \xrightarrow{e_{l-1}} r_{ss,l} \xrightarrow{e_1} r_{ss,1}$ satisfying condition (3). Let $\delta_{sa}(r_{ss,0}, w) = r_{ss,1}$. Since $r_{ss,j} = (x_{d,j}, \bar{x}_{d,j}) \in R_{sh}$ for some $j \in \{1, 2, \ldots, l\}$, from the construction of $SJ\text{-}Diag(G)$, it can be seen that $r_{ss,j} \in R_{sh}$ for all $j \in \{1, 2, \ldots, l\}$.

The cycle cl may correspond to a cycle $x_{d,1} \xrightarrow{e'_1} x_{d,2} \xrightarrow{e'_2} \cdots \xrightarrow{e'_{l-1}} x_{d,l} \xrightarrow{e'_1} x_{d,1}$ in $Diag_{att}(G)$, where $x_{d,j} = (x_j, F)$ for all $j \in \{1, 2, \ldots, l\}$. Let $\delta_{att}(x_{d,0}, w') = x_{d,1}$. By Proposition 1, it is $we_1e_2\cdots e_l = \xi(w'e'_1e'_2\cdots e'_l)$. By the assumption that there exists an event $e_j \in E_o \cup E_-$ in cl, associated with $w'e'_1e'_2\cdots e'_l$ is an arbitrarily long faulty word $\sigma_Y = \sigma t \in L(G)$, where $|t| \geq n$ for all $n \in \mathbb{N}$, such that $P(\sigma) = w'$ and $P(t) = e'_1e'_2\cdots e'_l$.

The cycle cl may also correspond to a cycle $\bar{x}_{d,1} \xrightarrow{e''_1} \bar{x}_{d,2} \xrightarrow{e''_2} \cdots \xrightarrow{e''_{l-1}} \bar{x}_{d,l} \xrightarrow{e''_1} \bar{x}_{d,1}$ in $Diag_{opr}(G)$. The following two cases could have occurred:

Case i): $\bar{x}_{d,1} = \bar{x}_{d,2} = \cdots = \bar{x}_{d,l}$. This means that $e_j \in E_-$ for all $j \in \{1, 2, \ldots, l\}$, since the occurrence of event $e_- \in E_-$ does not update state $\bar{x}_{d,j}$. Let $\delta_{opr}(\bar{x}_{d,0}, w'') = \bar{x}_{d,1}$. By Proposition 2, it holds that $w'' = \widehat{P}(we_1e_2\cdots e_l)$. Associated with w'' is a bounded normal word $\sigma_{N,1} \in L(G)$ such that $P(\sigma_{N,1}) = w''$. Hence, it is $P(\sigma_{N,1}) = w'' = \widehat{P}(we_1e_2\cdots e_l) = \widehat{P}(\xi(w'e'_1e'_2\cdots e'_l)) = \widehat{P}(\xi(P(\sigma t))) = \phi(P(\sigma_Y))$. By Definition 6, there exists an SH-attacker for G.

Case ii): In states $\bar{x}_{d,1}, \bar{x}_{d,2}, \cdots, \bar{x}_{d,l}$, there exist $n, m \in \{1, \cdots, l\}$ such that $\bar{x}_{d,n} \neq \bar{x}_{d,m}$. This means that $e_j \in E_o \cup E_+$ for some $j \in \{1, 2, \ldots, l\}$. Let $\delta_{opr}(x_{d,0}, w'') = \bar{x}_{d,1}$. By Proposition 2, it holds that $w''e''_1e''_2\cdots e''_l = \widehat{P}(we_1e_2\cdots e_l)$. If $\bar{x}_{d,j} = (x_j, N)$ for all $j \in \{1, 2, \ldots, l\}$, i.e., $\bar{x}_{d,j}$ are negative states, there exists an unbounded normal word $\sigma_{N,2} \in L(G)$ such that $P(\sigma_{N,2}) = w''e''_1e''_2\cdots e''_l$. Similar to Case i), it holds that $P(\sigma_{N,2}) = \phi(P(\sigma_Y))$. If $\bar{x}_{d,j} = (x_j, U)$ for all $j \in \{1, 2, \ldots, l\}$, i.e., $\bar{x}_{d,j}$ are uncertain states, as proved in [2], there also exists an unbounded normal word $\sigma_{N,3}$ such that $P(\sigma_{N,3}) = w''e''_1e''_2\cdots e''_l$. In addition, it holds that $\phi(P(\sigma_Y)) = P(\sigma_{N,3})$. Otherwise, the set $\bar{x}_{d,j}$ contains both normal and uncertain states, there also exists an unbounded normal word $\sigma_{N,4}$ such that $\phi(P(\sigma_Y)) = P(\sigma_{N,4}) = w''e''_1e''_2\cdots e''_l$. By Definition 6, the existence of σ_Y and $\sigma_{N,2}$ or $\sigma_{N,3}$ or $\sigma_{N,4}$ indicates the existence of an SH-attacker.

For both Cases i) and ii), since the SJD contains only stealthy attacks, the SH-attacker is also stealthy.

(Only if) Assuming in fact that there exists an SSH-attacker, the following situation must occur: (a) G can generate a word σ that ends with a fault event; there exists a normal word σ' satisfying $\phi(P(\sigma)) = P(\sigma')$; (b) the word σ can be extended indefinitely to a word $\sigma_k = \sigma e_1e_2\cdots e_k$ (for $k \geq 1$); there also exists a normal word $\sigma'_k \in L_N$ such that $\phi(P(\sigma_k)) = P(\sigma'_k)$ (for $k \geq 1$). Let the attack word $w = \xi(P(\sigma))$ reach a state $r_{ss,1}$ in $SJ\text{-}Diag(G)$. The following two cases could have occurred:

Case i): $r_{ss,1}$ is a harmful state with $d(r_{ss,1}) \in \{(F, N), (F, U)\}$). Starting from $r_{ss,1}$, as k grows, by Assumption A2, the attack word $w_k = \xi(P(\sigma_k))$ (for $k \geq 1$) also has an unbounded length and leads to a harmful state. Since the number of states of the SJD is finite, this is only possible if there exists a cycle of harmful states. Since $\sigma_k = uf e_1e_2\cdots e_k$ is of unbounded length, at least one event e in $E_o \cup E_-$ is contained along the harmful cycle.

Case ii): $r_{ss,1}$ is a non-harmful state with $d(r_{ss,1}) \in \{(U, N), (U, U)\}$): Starting from $r_{ss,1}$, as k grows, by Assumption A2, the attack word $w_k = \xi(P(\sigma_k))$ (for $k \geq 1$) of unbounded length is generated. However, according to Assumption A3 that $Diag(G)$ does not contain indeterminate cycles, every uncertain state will reach a positive state in $Diag(G)$ following $s_k = P(\sigma_k)$; correspondingly, the SJD will reach a harmful state by executing $w_k = \xi(P(\sigma_k))$ (for $k \geq 1$). Thus, the proof of Case ii) reduces to the case of i).

For both Cases i) and ii), the presence of an SSH-attacker implies that the SJD must contain cycles satisfying condition (3), which completes the proof. □

Different from the classical diagnosability verification by searching indeterminate cycles in a nominal diagnoser, the test of cycles in the SJD is self-contained. Namely, it does not require the examination of corresponding plant cycles. The reason is that once a harmful state is reached, a fault has certainly occurred previously (i.e., the diagnosis state of the attacker is "F"). As one continues along the cycle satisfying condition (3), the number of events generated after the fault increases indefinitely.

Finally, ignoring the attacker's stealthiness, the following result only guarantees its strong harmfulness, i.e., the impact on diagnosability, and serves as a solution to Problem 2 (that is a relaxation of Problem 4), whose proof is similar to that of Theorem 3 and thus omitted.

Corollary 2: Given a plant G, there exists an SH-attacker if and only if the JD $J\text{-}Diag(G)$ contains a cycle satisfying condition (3).

Example 7: Let us revisit the SJD of G in Fig. 1(a). Since $SJ\text{-}Diag(G)$ has a reachable cycle $(4F, 7N) \xrightarrow{b} (4F, 7N)$ satisfying condition (3) (highlighted by red lines), by Theorem 3, there exists an SSH-attacker for G. From $SJ\text{-}Diag(G)$, associated with the above cycle is an attack word $w = a_-b_-e_+db^k$, which is exactly produced by the

SSH-attacker ϕ described in Example 3. In addition, an inspection of $SJ\text{-}Diag(G)$ implies the attack actions of the attacker ϕ: first erasing event a and b and then inserting event e while observing $abdb^k$.

An SSH-attacker does not always exist. We consider a new plant H which is a copy of G in Fig. 1(a) but state 4 has a self-loop transition labeled with g. It can be verified that the JD $J\text{-}Diag(H)$ has a reachable cycle $(4F, 7N) \xrightarrow{b} (4F, 7N)$ satisfying condition (3). By Corollary 2, there exists an SH-attacker ϕ for H. Again consider the attack word $w = a\_b\_e_+db^k$. In $J\text{-}Diag(H)$, it leads to state $(4F, 7N)$. Using Eq. (2), we have $(4F, 7N) \in R_{we}$. Hence, the SH-attacker ϕ that produces $a\_b\_e_+db^k$ is no longer stealthy for H. In essence, if H infinitely generates the unerased event $g \in E_o \backslash E_{era}$ at state 4, it comes that $\phi(abdg) = edg \in L(\phi, H)$, but $\notin P(L(H))$. Finally, there is no SSH-attacker for H. $\diamond$

6 Conclusions and future work

In this paper, we have investigated the fault diagnosis in the case of attacks that corrupt sensor readings by inserting or erasing event observations. We formally formulate the problem of diagnosability under attack. From the attacker's point of view, not only its impact on the diagnosability but also its stealthiness should be taken into account. To this end, we propose a stealthy joint diagnoser (SJD), from which necessary and sufficient conditions for the existence of certain attackers related to diagnosability are presented.

Note that the proposed approach allows the integration with other common event-based techniques (e.g. supervisory control). To move forward, we further consider diagnosability enforcement if a system is no longer diagnosable due to the attack. Second, it is interesting to explore the use of verifier automata [3] or a polynomial diagnoser [12] for diagnosis and diagnosability verification under attack.

References

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

[2] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete-event systems," *IEEE Trans. Autom. Control*, vol. 40, no. 9, pp. 1555–1575, Sep. 1995.

[3] T. S. Yoo and S. Lafortune, "Polynomial-time verification of diagnosability of partially observed discrete-event systems," *IEEE Trans. Autom. Control*, vol. 47, no. 9, pp. 1491–1495, Sep. 2002.

[4] S. Takai, "A general framework for diagnosis of discrete event systems subject to sensor failures," *Automatica*, vol. 129, p. 109669, Jul. 2021.

[5] L. K. Carvalho, Y. C. Wu, R. Kwong, and S. Lafortune, "Detection and mitigation of classes of attacks in supervisory control systems," *Automatica*, vol. 97, pp. 121–133, Nov. 2018.

[6] Q. Zhang, C. Seatzu, Z. Li, and A. Giua, "Joint state estimation under attack of discrete event systems," *IEEE Access*, vol. 9, pp. 168 068–168 079, Dec. 2021.

[7] ——, "Selection of a stealthy and harmful attack function in discrete event systems," *Sci. Rep.*, vol. 12, no. 1, p. 16302, Sep. 2022.

[8] R. Meira-Góes, E. Kang, R. H. Kwong, and S. Lafortune, "Synthesis of sensor deception attacks at the supervisory layer of Cyber-Physical Systems," *Automatica*, vol. 121, p. 109669, Nov. 2020.

[9] Y. Li, C. N. Hadjicostis, and N. Wu, "Tamper-tolerant diagnosability under bounded or unbounded attacks." in *Proc. 16th Int. Workshop Discrete Event Syst.*, 2022, pp. 52–57.

[10] F. Lin, S. Lafortune, and C. Wang, "Diagnosability of discrete event systems under sensor attacks," in *Proc. 22th IFAC World Congr.*, 2023, pp. 32–38.

[11] T. Kang, C. Seatzu, Z. Li, and A. Giua, "Fault diagnosis of discrete event systems under attack," in *Proc. IEEE 62nd Conf. Decis. Control (CDC)*, 2023, pp. 7923–7929.

[12] F. G. Cabral, M. V. Moreira, O. Diene, and J. C. Basilio, "A Petri net diagnoser for discrete event systems modeled by finite state automata," *IEEE Trans. Autom. Control*, vol. 60, no. 1, pp. 59–71, Jan. 2015.