Secure Your Intention: On Notions of Pre-Opacity in Discrete-Event Systems

Shuo Yang, *Student Member, IEEE,* Xiang Yin, *Member, IEEE*

***Abstract*—This paper investigates an important information-flow security property called opacity in partially-observed discrete-event systems. We consider the presence of a passive intruder (eavesdropper) that knows the dynamic model of the system and can use the generated information-flow to infer some "secret" of the system. A system is said to be opaque if it always holds the plausible deniability for its secret. Existing notions of opacity only consider secret either as currently visiting some secret states or as having visited some secret states in the past. In this paper, we investigate information-flow security from a new angle by considering the secret of the system as the *intention* to execute some particular behavior of importance in the future. To this end, we propose a new class of opacity called *pre-opacity* that characterizes whether or not the intruder can predict the visit of secret states a certain number of steps ahead before the system actually does so. Depending the prediction task of the intruder, we propose two specific kinds of pre-opacity called K*-step instant pre-opacity* and K*-step trajectory pre-opacity* to specify this concept. For each notion of pre-opacity, we provide a necessary and sufficient condition as well as an effective verification algorithm. The complexity for the verification of pre-opacity is exponential in the size of the system as we show that pre-opacity is inherently PSPACE-hard. Finally, we generalize our setting to the case where the secret intention of the system is modeled as executing a particular sequence of events rather than visiting a secret state.**

***Index Terms*—Discrete-Event Systems, Opacity, Prediction**

I. Introduction

In the past decade, the notion of opacity has drawn a lot of attention in the Discrete-Event Systems (DES) literature as it provides a formal approach towards the verification and design of information-flow security for dynamic systems. Roughly speaking, opacity is a confidentiality property that captures whether or not the information-flow generated by a dynamic system can reveal some "secret behavior" to an outside observer (intruder) that is potentially malicious. In other words, an opaque system should always maintain the plausible deniability for its secret behavior during its execution. In the context of DES, opacity has been extensively studied for different system models including finite-state automata [1]–[4], labeled transition systems [5], [6] and Petri nets [7]–[10]. More recently, opacity has been extended to continuous dynamic systems with possibly infinite state spaces and time-driven dynamics [11]–[13]. Many enforcement techniques have also been proposed when the original system is not opaque; see, e.g., [14]–[24]. Opacity has also been applied to certify/enforce security in many real-world systems including mobile robots [25], location-based services [26], battery management systems [27] and web services [28]. The reader is referred to the survey papers [29], [30] for more details on opacity and its recent developments.

In order to capture different security requirements, different notions of opacity have been proposed in the literature. For example, in language-based opacity [31], the secret is formulated as the executions of some particular secret strings. As shown in [32], this formulation is equivalent to the notion of current-state opacity, where the secret is formulated as a set of secret states and a system is current-state opaque if the intruder cannot determine for sure that the system is currently at a secret state. In some situations, the system may want to hide its initial location or its location at some specific previous instant; such requirements can be captured by initial-state opacity [2] and K/infinite-step opacity [1], [3], [33], [34], respectively. More recently, quantitative notions of opacity have been proposed for stochastic DES in order to measure the secret leakage of the system; see, e.g., [35]–[40].

As we can see from the above discussion, "secret" in opacity analysis is actually a generic concept. Based on what kind of information the user would like to hide, or equivalently, how the intruder can utilize information to infer the secret of the system, existing notions of opacity in the literature as reviewed above can generally be divided into the following two categories:

- Opacity for Current Information: the intruder wants to determine the current behavior of the system based on the current observation. In other words, the user does not want the outsider to know for sure that it *is currently doing* something secret. This category includes, e.g., current-state opacity and language-based opacity.
- Opacity for Delayed Information: the intruder wants to determine the previous secret behavior of the system at some instant based on the current observation. In other words, the user does not want the outsider to know for sure that it *has done* something secret at some previous instant. This category includes, e.g., initial-state opacity, K-step opacity and infinite-step opacity. Note that delayed information is involved here as the intruder does not need to specify the visit of a secret state immediately; it can use future information to improve its knowledge about the previous instants.

This work was supported by the National Natural Science Foundation of China (62061136004, 62173226, 61833012) and the National Key Research and Development Program of China (2018AAA0101700).

Shuo Yang is with Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA, yangs1@seas.upenn.edu,

Xiang Yin is with Department of Automation and Key Laboratory of System Control and Information Processing, Shanghai Jiao Tong University, Shanghai 200240, China, yinxiang@sjtu.edu.cn. (Corresponding Author: Xiang Yin)

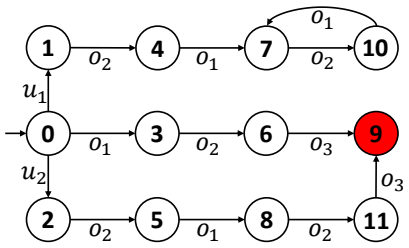


Fig. 1. A motivating example with $E_o = \{o_1, o_2, o_3\}$ and $E_{uo} = \{u_1, u_2\}$. State 9 is the target (secret) state.

There are also some works that combine these two types of opacity together, e.g., by combing current-state opacity and initial-state opacity, one can define the notion of initial-final-state opacity [32].

In some applications, however, the "secret" one wants to hide can be its *intention* to do something of particular importance in the future. As a simple motivating example, let us consider a single robot moving in a region whose mobility is described by a DES shown in Figure 1, where each state represents a location and each transition represents an action. Some actions are assumed to be observable by outsider; $E_o = \{o_1, o_2, o_3\}$ are observable actions. The robot may choose to attack state 9 by reaching it. However, it does not want to reveal its intention to attack state 9 too early; otherwise, e.g., some defense strategy can be implemented in advance. Clearly, the shortest path to reach state 9 is $0 \xrightarrow{o_1} 3 \xrightarrow{o_2} 6 \xrightarrow{o_3} 9$. However, by doing so, the outsider will know the robot's intention of attack two steps ahead just by observing the first action o_1. On the other hand, the robot can choose to attack state 9 via path $0 \xrightarrow{u_2} 2 \xrightarrow{o_2} 5 \xrightarrow{o_1} 8 \xrightarrow{o_2} 11 \xrightarrow{o_3} 9$, which is longer but allows the robot to hide its intention of visiting state 9 until it actually reaches it. This is because this path has the same observation of $0 \xrightarrow{u_1} 1 \xrightarrow{o_2} 4 \xrightarrow{o_1} 7 \xrightarrow{o_2} 10$ whose continuation may not necessarily be secret. Existing notions of opacity cannot capture this scenario as this problem essentially requires another type of opacity for *future information*: the user does not want the outsider to know too early for sure that it *will do* something secret at some future instant.

In this paper, we investigate opacity from a new angle by considering the system's *intention* of executing some particular behavior as the secret. Then we propose a new type of opacity, called *pre-opacity*, to characterize whether or not the secret intention of the system can be revealed. We follow the standard setting of opacity by considering a passive intruder modeled as an eavesdropper that knows the model of the system. Then we propose two notions of pre-opacity called *K-step instant pre-opacity* and *K-step trajectory pre-opacity*; both require that the intruder cannot determine secret future information K-step ahead. However, the former emphasizes that the intruder cannot determine the *specific* instant when the system will be at secret states, while the latter requires that the intruder cannot determine that the system will visit secret states in the future unavoidably without the need of specifying the precise instant of being secret. Properties of these two notions of pre-opacity are investigated and we show that instant pre-opacity is strictly weaker than trajectory pre-opacity. Furthermore, for each pre-opacity, we provide necessary and sufficient condition as well as effective verification algorithm. We show that both properties are PSPACE-hard; hence the exponential verification complexity is unavoidable. Also, we discuss the case where "secrets" are modeled as a *sequence pattern* rather secret states.

In the systems theory, there are three fundamental types of estimation problems: filtering, smoothing and prediction. Essentially, current-state opacity can be viewed as the plausible deniability for secret under filtering and infinite/K-step opacity can be viewed as the plausible deniability for secret under smoothing. Analogously, the proposed notion of pre-opacity can also be interpreted as the plausible deniability for secret under prediction. Therefore, our new notion also generalizes the framework of opacity from the systems theory point of view.

The proposed notion of pre-opacity, in particular, trajectory pre-opacity, is closely related to the notion of fault predictability (or prognosability) in the literature; see, e.g., [41]–[46]. However, predictability requires that any fault can be predicted before its occurrence, but our notion of pre-opacity requires that any secret cannot be predicted before it actually happens. Furthermore, our notion of instant pre-opacity is much more different since it requires to determine the precise instant of being secret, which is not required in predictability analysis. Also, in fault prediction problems, one is only interested in predicting the first occurrence of fault. However, in pre-opacity analysis, the system's behavior can become secret/non-secrete intermittently in the sense that, even when the intruder fails to predict the first secret behavior, it may still has chance to predict some future secret so that the security of the system can still be threatened. Therefore, although predictability is conceptually related to our notion of pre-opacity, these two properties are technically very different.

The rest of the paper is organized as follows. In Section II, we describe the system model and review the existing notions of opacity. Section III introduces the two new notions of pre-opacity and discusses their properties. In Section IV, we provide effective algorithms for the verification of notions of pre-opacity. The proposed pre-opacity is further generalized to the case of sequence pattern in Section V. Finally, we conclude this paper by Section VI.

II. Preliminaries

A. System model

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

We consider a discrete-event system modeled by a finite-state automaton (FSA)

$$G = (X, E, f, X_0, X_m),$$

where X is the finite set of states, E is the finite set of events, $f : X \times E \to X$ is the partial deterministic transition function

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2022.3210148

such that $f(x,\sigma) = x'$ means that there exists a transition from x to x' with event label σ, $X_0 \subseteq X$ is the set of initial states and $X_m \subseteq X$ is the set of marked states. The transition function f is also extended to $f : X \times E^* \to X$ recursively by: for any $x \in X, s \in E^*, \sigma \in E$, we have $f(x, s\sigma) = f(f(x,s),\sigma)$ with $f(x,\epsilon) = x$.

The language generated by G from state $x \in X$ is defined by $\mathcal{L}(G,x) = \{s \in E^* : f(x,s)!\}$, where "!" means "is defined". Also, we define $\mathcal{L}(G,Q) := \bigcup_{x\in Q} \mathcal{L}(G,x)$ as the language generated from a set of states $Q \subseteq X$. Therefore, the language generated by G is $\mathcal{L}(G) := \mathcal{L}(G, X_0)$. For the sake of simplicity, hereafter, we assume that the system G is live, i.e., for any $x \in X$, there exists $\sigma \in \Sigma$ such that $f(x,\sigma)!$. Then the marked language of G is $\mathcal{L}_m(G) = \{s \in E^* : \exists x_0 \in X_0, \text{ s.t. } f(x_0,s) \in X_m\}$. When marked states are not considered, i.e., $X_m = \emptyset$, we will omit X_m from the tuple and represent a FSA by $G = (X, E, f, X_0)$.

B. Intruder Model and Opacity

Following the standard setting of opacity, we assume that the intruder is modeled as a *passive observer* (eavesdropper), which has the full knowledge of the system's structure. By "passive", we mean that the intruder can only observe some behavior generated by the system, but it cannot actively affect the behavior of the system. Formally, we assume that the event set E is partitioned as:

$$E = E_o \dot\cup E_{uo},$$

where E_o and E_{uo} are the set of observable events and the set of unobservable events, respectively. The natural projection from E to E_o is a mapping $P : E^* \to E_o^*$ defined recursively by:

$$P(\epsilon) = \epsilon \quad \text{and} \quad P(s\sigma) = \begin{cases} P(s)\sigma & \text{if } \sigma \in E_o \\ P(s) & \text{if } \sigma \notin E_{uo} \end{cases} \tag{1}$$

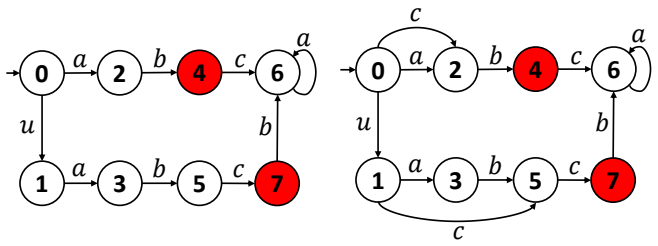
The natural projection is also extended to $P : 2^{E^*} \to 2^{E_o^*}$, i.e., $P(L) = \{t \in E_o^* : \exists s \in L \text{ s.t. } P(s) = t\}$ for any $L \subseteq E^*$.

When string $s \in \mathcal{L}(G)$ is generated by the system, the intruder observes $P(s)$ and it can use this observation together with the dynamic model of the system to infer which state the system could be in at some specific instant. In opacity analysis, it is assumed that the system has a set of secret states, denoted by $X_S \subseteq X$. Roughly speaking, a system is said to be opaque if the intruder can never determine for sure that the system is/was at a secret state based on its observation. Here, we review the notion of K-step opacity which can be used to define current-state opacity and infinite-step opacity.

Definition 1. *(K-Step Opacity) [1] Given system G, set of observable events E_o, set of secret states X_S, and non-negative integer $K \in \mathbb{N}$, system G is said to be* K-step opaque *(w.r.t. E_o and X_S) if*

$$\begin{aligned} &(\forall x_0 \in X_0, \forall st \in \mathcal{L}(G,x_0) : f(x_0,s) \in X_S \wedge |P(t)| \le K) \\ &(\exists x_0' \in X_0)(\exists s't' \in \mathcal{L}(G,x_0')) \text{ s.t.} \\ &[P(s) = P(s')] \wedge [P(t) = P(t')] \wedge [f(x_0',s') \notin X_S]. \end{aligned} \tag{2}$$

Furthermore, system G is said to be



(a) G_1 (b) G_2

Fig. 2. For both systems, we have $X_0 = \{0\}, X_S = \{4,7\}$ and $E_o = \{a,b,c\}$.

- current-state opaque *if it is* 0*-step opaque;*
- infinite-step opaque *if it is K-step opaque for any $K \ge 0$.*

Intuitively, K-step opacity says that, whenever the system visits a secret state, it should be able to keep this secret unrevealed within the next K steps. In other words, the intruder should never be able to determine that the system was at a state secret for any instant in the past K steps. Note that current-state opacity can be viewed as a special case of K-step opacity ($K = 0$) as it essentially requires that the intruder cannot determine for sure that the system is currently at a secret state. To verify current-state opacity, one can construct the current-state estimator (or the observer) and check whether or not there exists a reachable estimator state that only contains secret states. The verification of K-step opacity and infinite-step opacity are more involved as they require the computation of delayed state estimate, which can be done by constructing the two-way observer [3].

Example 1. *Let us consider system G_1 shown in Figure 2(a), where $X_S = \{4,7\}$ and $E_o = \{a,b,c\}$. Clearly, this system is current-state opaque. For example, by observing ab, the intruder cannot determine whether the system is at secret state 4 or at non-secret state 5 since $P(ab) = P(uab) = ab$. Similarly, when secret state 7 is reached via $uabc$, the intruder still cannot distinguish this state from non-secret state 6. On the other hand, this system is not 1-step opaque. This is because, by observing $abcb$, the intruder can determine for sure that the system was at secret state 7 one step ago. Therefore, G_1 is also not infinite-step opaque.*

III. Notions of Pre-Opacity

In this section, we first provide the definitions of K-step instant pre-opacity and K-step trajectory pre-opacity for DES. Then we discuss properties of the proposed notions of pre-opacity.

A. Definitions of K-step Instant/Trajectory Pre-Opacity

In the existing notions of opacity, secret is either characterized by whether the system is doing something secret (current-state opacity) or characterized by whether the system has done something secret (K-step and infinite-step opacity). These settings essentially assume that the system is operating against an intruder whose functionality is a current-state estimator or a delayed state-estimator.

However, in some applications, what the system wants to hide might be its *intention*, i.e., maintain the plausible deniability for its willing to do something secret in the future. In this setting, the system is essentially operating against an intruder that can be interpreted as a *predictor*. More specifically, the user may require that the intruder should never be able to determine its intention of visiting a secret state too early, which is characterized by a parameter K. To this end, we first propose the notion of *K-step instant pre-opacity* as follows; the reason why we use terminology "instant" here will be clear soon.

Definition 2. *(K-Step Instant Pre-Opacity) Given system G, set of observable events E_o, set of secret states X_S, and non-negative integer $K \in \mathbb{N}$, system G is said to be K-step instant pre-opaque (w.r.t. E_o and X_S) if*

$$\begin{aligned}&(\forall x_0 \in X_0, \forall s \in \mathcal{L}_o(G, x_0))(\forall n \geq K)\\ &(\exists x_0' \in X_0, \exists s' \in \mathcal{L}_o(G, x_0'), \exists t \in \mathcal{L}(G, f(x_0', s'))\ \textit{s.t.}\\ &[P(s) = P(s')] \wedge [|t| = n] \wedge [f(x_0', s't) \notin X_S] \end{aligned} \tag{3}$$

where

$$\mathcal{L}_o(G, x) := (\mathcal{L}(G, x) \cap E^* E_o) \cup \{\epsilon\}$$

is the set of strings generated from x that end up with observable events including the empty string.

Intuitively, K-step instant pre-opacity requires that, for any string s generated from some initial state x_0 and any future instant $n \geq K$, there exists another observation-equivalent string s' generated from some initial state x_0' such that s' can reach a non-secret state in exact n steps. In other words, the intruder can never determine more than K steps ahead, based on its current observation, that the system will visit a secret state at some future instant. Therefore, K can be viewed as a parameter that determines *how early* the user does not want to reveal its intention. For instance, if $K = 2$, then the user may allow the intruder to determine just one step ahead that it will visit a secret system. We use the following example to illustrate this notion.

Example 2. *First, let us consider again system G_1 in Figure 2(a). One can easily check that this system is 1-step instant pre-opaque. For example, for string $a \in \mathcal{L}_o(G)$, the intruder cannot predict for sure that the system will be at a secret in one step since there exist another string ua and its one-step extension b such that $P(ua) = P(a)$ but $f(0, uab) = 5 \notin X_S$. Similarly, the intruder also cannot predict for sure that the system will be at a secret after 2 steps. For example, when observing ϵ, the system may reach non-secret state 3 in two steps, which protects the possible secret intention of executing ab; when observing a, the system may reach non-secret state 6 in two step, which protects the possible secret intention of executing $uabc$.*

However, for system G_2 in Figure 2(b), where $X_S = \{4, 7\}$ and $E_o = \{a, b, c\}$, one can check that this system is not 1-step instant pre-opaque. This is because, by observing c, the intruder can determine for sure that the system is either at state 2 or at state 5. However, from either state 2 or 5, the system will reach a secret state in the next step. Therefore, its intention of visiting secret states will be revealed one step before it actually happens.

Remark 1. *In Definition 2, "step" is counted by the number of occurrences of* actual events *rather than the occurrences of observable events, i.e., we consider $|t| = n$ rather than $|P(t)| = n$. We believe this setting is more natural for predicting future instants. Furthermore, we consider string s in $\mathcal{L}_o(G, x_0)$ rather than $\mathcal{L}(G, x_0)$. This implicitly assumes that the intruder will make a prediction immediately after observing a new observable event. Hereafter, we will introduce the main developments based on this setting.*

Note that K-step instant pre-opacity requires that the intruder cannot predict K-step ahead that the system will visit a secret state at some *specific instant*. This is also why we call it "instant" pre-opacity. However, in some situations, the intruder may just want to know whether or not the system will visit a secret state in the future without the need of telling the specific instant. For instance, for G_2 in Figure 2(a), after observing a, although the intruder cannot determine for sure the specific instant when the secret state will be reached (the system will visit a secret state in one step or in two steps), it can still tell that the system will visit a secret state within the next two steps and at least one step before the occurrence of the first secret state. To capture this scenario, we propose the notion of K-step trajectory pre-opacity.

Definition 3. *(K-Step Trajectory Pre-Opacity) Given system G, set of observable events E_o, set of secret states X_S, and non-negative integer $K \in \mathbb{N}$, system G is said to be K-step trajectory pre-opaque (w.r.t. E_o and X_S) if*

$$\begin{aligned}&(\forall x_0 \in X_0, \forall s \in \mathcal{L}_o(G, x_0))(\forall n \geq K)\\ &(\exists x_0' \in X_0, \exists s' \in \mathcal{L}_o(G, x_0'), \exists t_1 t_2 \in \mathcal{L}(G, f(x_0', s'))\ \textit{s.t.}\\ &[P(s) = P(s')] \wedge [|t_1| = K] \wedge [|t_1 t_2| = n] \wedge\\ &\quad [\forall w \in \overline{\{t_2\}} : f(x_0', s't_1 w) \notin X_S] \end{aligned}$$

Intuitively, K-step trajectory pre-opaque says that the intruder will never be able to determine K-step ahead for sure that the system will visit a secret state. More specifically, if a system is not K-step trajectory pre-opaque, then according to Definition 3, it means that there exist a string s such that any observation equivalent string s' must pass through a secret state after the next Kth instant. In other words, the intruder can determine the system's intention of visiting a secret state more than K-step ahead. We use the following example to illustrate this notion.

Example 3. *Let us consider again G_1 shown in Figure 2(a) and we have shown in Example 2 that this system is 1-step instant pre-opaque. However, it is not 1-step trajectory pre-opaque. For example, let us consider $\epsilon \in \mathcal{L}_o(G)$ and $n = 4 \geq 1 = K$. Note that ϵ itself is the only observation equivalent string in $\mathcal{L}_o(G)$. However, any 4-step extension of ϵ, either $abca$ or $uabc$, will necessarily pass through a secret state between the first instant and the forth instant. On the other hand, this system is 3-step trajectory pre-opaque. This is because the only instant to predict the visit of a secret state 3-step ahead is when observing ϵ. However, with this*

observation, it is possible that the system will be at state 6*, from which no secret state will be visited, after three steps. Therefore, the intruder can never determine* 3*-step ahead for sure that a secret state will be visited.*

Remark 2. *Similar to the interpretations of current-state opacity and K-step opacity, where the system is operating against the current-state estimator and delay-state estimator, respectively, here one can image that the system is operating against an intruder working as a* predictor *(for its secret intention). Roughly speaking, both K-step instant pre-opacity and K-step trajectory pre-opacity require that the intruder can never predict its secret K-steps ahead. However, the specific prediction tasks of the "virtual predictor" in these two notions are different: in instant pre-opacity, the predictor also needs to identify the precise future instant at which the system will be at a secret state, while in trajectory pre-opacity, the predictor just needs to identify the inevitability of passing through a secret state without specifying the visiting instant.*

B. Properties of Pre-Opacity

Now, we discuss properties of the proposed notions of pre-opacity and their relationships with other notions of opacity in the literature. First, we show that, for any K, K-step instant pre-opacity is weaker than K-step trajectory pre-opacity.

Proposition 1. *If G is K-step trajectory pre-opaque, then it is K-step instant pre-opaque.*

Proof. This result follows directly from the definitions. If the system is K-step trajectory pre-opaque, then by setting t in Definition 2 as t_1t_2 in Definition 3, we know that the system is K-step instant pre-opacity. □

The intuition of the above result can also be interpreted as follows. For the case of instant pre-opacity, the prediction task of intruder is more challenging than that for the case of trajectory opacity due to the need of determining the specific secret instant. Therefore, from the system's point of view, the underlying security property becomes weaker.

Also, by definitions, we note that K-step instant pre-opacity becomes weaker when K increases, i.e., K-step instant pre-opacity always implies $(K+1)$-step instant pre-opacity. However, the following result shows that there is an upper bound for K in instant pre-opacity, i.e., pre-opacity will not keep getting strictly weaker when K increases.

To present our result, we introduce two necessary concepts. First, for each state $x \in X$, the set of states that can be reached from x in exactly K steps is define by

$$R_K(x) = \{x' \in X : \exists s \in \mathcal{L}(G,x) \text{ s.t. } f(x,s) = x' \wedge |s| = K\}. \tag{4}$$

For a set of states $q \subseteq X$, we also define $R_K(q) := \bigcup_{x\in q} R_K(x)$ as the set of states that can be reached from set q in exactly K steps.

Also, let $\alpha \in P(\mathcal{L}(G))$ be an observed string. Then the current-state estimate upon the occurrence of α without the unobservable tail is defined by

$$\hat{\mathcal{E}}(\alpha) = \{f(x_0,s) \in X : \exists x_0 \in X_0, s \in \mathcal{L}_o(G,x_0) \text{ s.t. } P(s) = \alpha\}. \tag{5}$$

Then we have the following the theorem showing the upper bound of K in instant pre-opacity.

Theorem 1. *For any $K' > K \geq 2^{|X|} - 1$, system G is K'-step instant pre-opaque, if and only if, G is K-step instant pre-opaque.*

Proof. It is trivial that K-step instant pre-opacity implies K'-step instant pre-opacity. Hereafter, we show that K'-step instant pre-opacity also implies K-step instant pre-opacity. Without loss of generality, we assume that $K' = K+1$ as the argument can be applied inductively.

Now we assume, for the sake of contradiction, that G is not K-step instant pre-opaque but G is $(K+1)$-step instant pre-opaque, where $K \geq 2^{|X|} - 1$. This implies that there exists an initial state $x_0 \in X_0$ and a string $s \in \mathcal{L}_o(G,x_0)$ such that

$$\begin{aligned}&(\forall x_0' \in X_0)(\forall s' \in \mathcal{L}_o(G,x_0'), s't \in \mathcal{L}(G,x_0'))\\ &[P(s) = P(s') \wedge |t| = K] \Rightarrow [f(x_0', s't) \in X_S].\end{aligned}$$

Equivalently, we have $R_K(\hat{\mathcal{E}}(P(s))) \subseteq X_S$. Since for any $i \in \mathbb{N}$, $R_i(\hat{\mathcal{E}}(P(s)))$ is non-empty and it has at most $|X|$ elements, there are only $(2^{|X|}-1)$ choices for $R_i(\hat{\mathcal{E}}(P(s)))$. Moreover, since the cardinality of multi-set $\{R_j(\hat{\mathcal{E}}(P(s))) : j = 0,1,\cdots,K\}$ is $K+1 \geq 2^{|X|} > 2^{|X|}-1$, we know that there exist two integers $0 \leq m < n \leq K$, such that $R_m(\hat{\mathcal{E}}(P(s))) = R_n(\hat{\mathcal{E}}(P(s)))$. Then we know that

$$\begin{aligned}R_{K+n-m}(\hat{\mathcal{E}}(P(s))) &= R_{K-m}(R_n(\hat{\mathcal{E}}(P(s))))\\ &= R_{K-m}(R_m(\hat{\mathcal{E}}(P(s)))) = R_K(\hat{\mathcal{E}}(P(s))) \subseteq X_S\end{aligned} \tag{6}$$

i.e., for initial state $x_0 \in X_0$ and string $s \in \mathcal{L}_o(G,x_0)$, we also have that

$$\begin{aligned}&(\forall x_0' \in X_0)(\forall s' \in \mathcal{L}_o(G,x_0'), s't \in \mathcal{L}(G,x_0'))\\ &[P(s) = P(s') \wedge |t| = K+n-m] \Rightarrow [f(x_0', s't) \in X_S].\end{aligned}$$

Since $K+n-m \geq K+1$, we know that $(K+1)$-step instant pre-opacity is violated, which is a contradiction. □

One may conjecture that 0-step instant pre-opacity is equivalent to current-state opacity. However, it is not exactly the case. For example, let us consider system G_3 shown in Figure 4. This system is current-state opaque as the intruder cannot distinguish states 1 and 2 after observing a due to unobservable event u. On the other hand, it is not 0-step instant pre-opaque according to our definition since the intruder can predict one step ahead for sure that the system will reach the secret state when observing nothing. This difference is due to the fact that we consider instant in terms of actual event steps rather than the observation steps. The only conclusion we can draw is that current-state opacity is weaker than 0-step instant pre-opacity, which is stated as follows.

Proposition 2. *If G is 0-step instant/trajectory pre-opaque, then G is current-state opaque.*

Proof. It suffices to show that 0-step instant pre-opacity implies current-state opacity since 0-step trajectory pre-opacity is stronger than 0-step instant pre-opacity. Suppose G is 0-step instant pre-opaque. Let us consider arbitrary initial state $x_0 \in X_0$ and string $s \in \mathcal{L}(G,x_0)$. Note that for string s, we

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2022.3210148

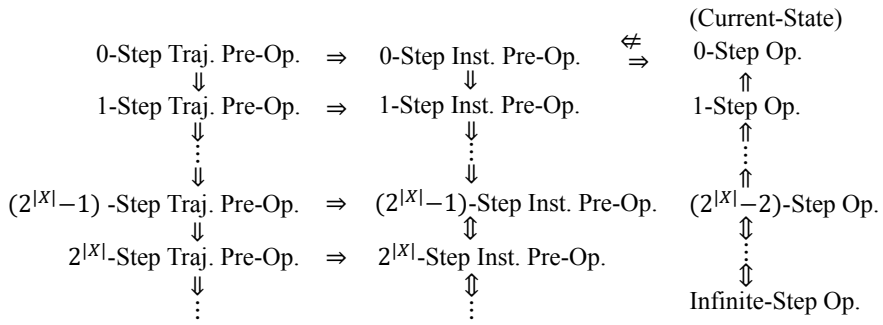


Fig. 3. Relationships among different notions of opacity and pre-opacity.

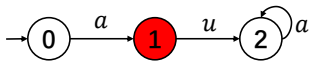


Fig. 4. A system that is current-state opaque but is not 0-step instant pre-opaque, where $X_S = \{1\}$ and $E_{uo} = \{u\}$.

can always find $\hat{s} \in \mathcal{L}_o(G, x_0)$, by removing the unobservable tail (if any) of s such that $P(s) = P(\hat{s})$. Since G is 0-step instant pre-opaque, by setting n in Definition 2 as $n = 0$, we know that there exist $x_0' \in X_0$ and $s' \in \mathcal{L}_o(G, x_0'))$ such that $P(s') = P(\hat{s}) = P(s)$ and $f(x_0', s') \notin X_S$. This means that the system is current-state opaque. □

Based on the above discussion, we summarize the relationships among the proposed notions of pre-opacity and existing notions of opacity in Figure 3.

Remark 3. *Finally, we note that the proposed concept of pre-opacity is also related to the notion of fault predictability or fault prognosability in the literature [41]–[43], which captures whether or not a fault event can always be predicted unambiguously a certain number of steps ahead before it actually occurs. Conceptually, by considering the visit of secret states as fault, trajectory pre-opacity can be viewed as a dual problem of predictability. However, trajectory pre-opacity is not exactly non-predictability. The former requires that* all *secret paths cannot be predicted, while the latter says* some *fault path cannot be predicted. Furthermore, our notion of instant opacity is quite different from predictability as we need to determine the specific instant of being secret; this issue does not occur in predictability analysis.*

IV. Verification of Pre-Opacity

In this section, we show how to verify the proposed notions of pre-opacity. Specifically, we present two state-based necessary and sufficient conditions for K-step instant pre-opacity and K-step trajectory pre-opacity, respectively, that can be checked using the observer structure. Then we discuss the complexity of the verification problems.

A. Necessary and Sufficient Condition for Instant Pre-Opacity

Recall that a system is not K-step instant pre-opaque if after some observation, each possible state (immediately after the observation) will visit a secret state in exactly n steps for some $n \geq K$. This suggests that K-step instant pre-opacity can be checked by combining the current-state estimation together with the reachability analysis. To this end, we further introduce some necessary notions.

We say that a state $x \in X$ is a *K-step indicator* state if it will reach a secret state inevitably in exactly K steps, i.e.,

$$R_K(x) \subseteq X_S.$$

For any $K \in \mathbb{N}$, we define

$$\Im_K := \{x \in X : R_K(x) \subseteq X_S\} \subseteq X$$

as the set of K-step indicator states.

Then the following theorem shows that K-step instant pre-opacity can be simply characterized in terms of current-state estimate and K-step indicator states.

Theorem 2. *System G is K-step instant pre-opaque if and only if*

$$\forall \alpha \in P(\mathcal{L}(G)), \forall n \geq K : \hat{\mathcal{E}}(\alpha) \not\subseteq \Im_n.$$

Proof. ($\Rightarrow$) By contraposition. Suppose that there exists a string $\alpha \in P(\mathcal{L}(G))$ and an integer $n \geq K$ such that $\hat{\mathcal{E}}(\alpha) \subseteq \Im_n$. Let us consider an initial state $x_0 \in X_0$ and a string $s \in \mathcal{L}_o(G, x_0)$ such that $P(s) = \alpha$. Since $\hat{\mathcal{E}}(\alpha) \subseteq \Im_n$, for any initial state $x_0' \in X_0$ and string $s' \in \mathcal{L}_o(G, x_0)$ such that $P(s') = \alpha$, we have $f(x_0', s') \in \Im_n$, i.e., $R_n(f(x_0', s')) \subseteq X_S$ This means that for for any $t \in \mathcal{L}(G, f(x_0', s'))$ and $|t| = n$, we have $f(x_0', s't) \in X_S$. This means that system G is not K-step instant pre-opaque.

($\Leftarrow$) Still by contraposition. Suppose that G is not K-step instant pre-opaque, which means that there exists an initial state $x_0 \in X_0$, a string $s \in \mathcal{L}_o(G, x_0)$ and an integer $n \geq K$ such that

$$(\forall x_0' \in X_0)(\forall s' \in \mathcal{L}_o(G, x_0'), s't \in \mathcal{L}(G, x_0'))$$
$$[P(s) = P(s') \wedge |t| = n \Rightarrow f(x_0', s't) \in X_S]$$

Then let $\alpha = P(s)$. Clearly, we have $R_n(\hat{\mathcal{E}}(\alpha)) \subseteq X_S$, i.e., $\hat{\mathcal{E}}(\alpha) \subseteq \Im_n$. This violates the condition in the theorem. □

Theorem 2 essentially provides a state-based characterization of the language-based definition of K-step instant pre-opacity. However, it still cannot be directly used for the

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2022.3210148

verification of instant pre-opacity. The main issue is that we need to check whether or not $\hat{\mathcal{E}}(\alpha) \not\subseteq \Im_n$ for any $n \geq K$, which has infinite number of instants. The following result further generalizes Theorem 2 and shows that it suffices to check $\hat{\mathcal{E}}(\alpha) \not\subseteq \Im_n$ for a bounded number of instants.

Proposition 3. *For any $\alpha \in P(\mathcal{L}(G))$, the following two statements are equivalent:*

(i) $\forall n \geq K : \hat{\mathcal{E}}(\alpha) \not\subseteq \Im_n$;
(ii) $\forall n \in \{K, K+1, \ldots, K+2^{|X|}-1\} : \hat{\mathcal{E}}(\alpha) \not\subseteq \Im_n$.

Proof. (i)⇒(ii) is trivial. Hereafter, we show that (ii)⇒(i). Let $q := \hat{\mathcal{E}}(\alpha)$ and we consider the reachable set of q for each instant between K and $K+2^{|X|}-1$, i.e., $R_K(q), R_{K+1}(q), \ldots, R_{K+2^{|X|}-1}(q)$. For any $n \in \{K, \ldots, K+2^{|X|}-1\}$, since $q \not\subseteq \Im_n$, we know that there exists $x \in q$ such that $x \notin \Im_n$, i.e., $R_n(x) \not\subseteq X_S$. Since $R_n(q) = \bigcup_{x \in q} R_n(x)$, we know that $R_n(q) \not\subseteq X_S$ for any $n \in \{K, \ldots, K+2^{|X|}-1\}$.

Now we note that set $\{R_i(q) : K \leq i \leq K+2^{|X|}-1\} \subseteq 2^X$ is non-empty, so it contains at most $2^{|X|}$ elements. Therefore, there must exist two instants $K \leq i < j \leq K+2^{|X|}-1$ such that $R_i(q) = R_j(q)$. Furthermore, by the definition of K-step reachable set, we have

$$R_{n+k}(q) = R_n(R_k(q)) = R_k(R_n(q))$$

Then for any instant $n' > K+2^{|X|}-1$, we can always write it in the form of

$$n' = i + (j-i) \times k + m$$

where $1 \leq k, 0 \leq m < (j-i)$ are two integers. Furthermore, since $R_i(q) = R_j(q)$, we have $R_i(q) = R_{i+(j-i)\times k}(q)$ for any $k \geq 0$, so

$$R_{n'}(q) = R_m(R_{i+(j-i)\times k}(q)) = R_{m+i}(q).$$

However, since $m < j-i$, we have $m+i < j$. Therefore,

$$\forall n' > K+2^{|X|}-1 : R_{n'}(q) = R_{m+i}(q) \not\subseteq X_S.$$

This further implies that

$$\forall n' > K+2^{|X|}-1 : q \not\subseteq \Im_{n'},$$

which completes the proof. □

One may ask why we need to search for the entire next $2^{|X|}$ instants to obtain the upper bound in Proposition 3. However, this upper bound seems to be unavoidable. To see this, let us consider the system shown in Figure 5, where all events are unobservable and red states denote secret states. This system is not K-step instant pre-opaque for any K since one can determine for sure (by observing nothing) that the system will be at a secret state for instants $k \cdot 30, k = 1, 2 \ldots$, where 30 is the least common multiple of cycle lengths $2, 3$ and 5. Therefore, the first violation of $\hat{\mathcal{E}}(\alpha) \not\subseteq \Im_n$ occurs at $n = 30$. Similarly, one could add more states to create more such cycles and the upper bound for searching $\Im_n$ will grow exponentially. However, this exponentially searching bound is only needed when the system contains unobservable events. In the following result, we show that such an upper bounded search can be avoided for the extreme case when there is no unobservable event in the system.

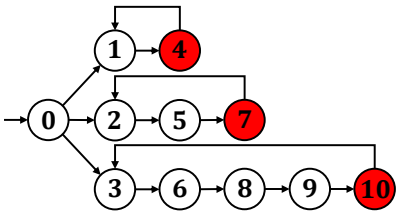


Fig. 5. A system where all events are unobservable and red states denote secret states.

Proposition 4. *Under the assumption that all events in G are observable, then G is K-step instant pre-opaque if and only if*

$$\forall \alpha \in P(\mathcal{L}(G)) : \hat{\mathcal{E}}(\alpha) \not\subseteq \Im_K.$$

Proof. The necessity follows directly from Theorem 2. To show the sufficiency, suppose that $\forall \alpha \in P(\mathcal{L}(G)) : \hat{\mathcal{E}}(\alpha) \not\subseteq \Im_K$ and assume that G is not K-step instant pre-opaque. Then by Theorem 2, we know that there exist $\alpha \in P(\mathcal{L}(G))$ and $n > K$ such that $\hat{\mathcal{E}}(\alpha) \subseteq \Im_n$. In other words, we have that

$$(\forall x_0 \in X_0)(\forall s \in \mathcal{L}_o(G, x_0), st \in \mathcal{L}(G, x_0))$$
$$[P(s) = \alpha \wedge |t| = n] \Rightarrow [f(x_0, st) \in X_S]$$

For any t satisfying above condition, we let $t = t_1 t_2$, where $|t_1| = n - K$ and $|t_2| = K$. Then we know that $R_K(\hat{\mathcal{E}}(P(st_1))) \subseteq X_S$, i.e., $\hat{\mathcal{E}}(P(st_1)) \subseteq \Im_K$, which is a contradiction. □

B. Necessary and Sufficient Condition for Trajectory Pre-Opacity

Now we discuss the case of K-step trajectory pre-opacity. First, we say that a state $x \in X$ is a *non-indicator state* if there exists an infinitely long string defined from this state along which no secret state is visited. Formally, we define the set of non-indicator states by

$$\mathcal{N} := \left\{ x \in X : \begin{array}{c} (\forall n \geq 0)(\exists s \in \mathcal{L}(G, x) : |s| > n) \\ (\forall t \in \overline{\{s\}})[f(x, t) \notin X_S] \end{array} \right\} \quad (7)$$

Since the number of states in G is finite, a state is a non-indicator state if and only it can reach a cycle, in which all states are non-secret, via a sequence of non-secret states. In other words, if a state is not in $\mathcal{N}$, then it is an indicator state in the sense that a secret state will be visited inevitably from this state.

Remark 4. *Note that a state is not a non-indicator state does not necessarily imply that it is a K-step indicator state for some K since the latter requires the system to be at a secret state for some specific instant while the former does not require this information. Furthermore, a state is an indicator state does not imply that any state reachable from this state is an indicator state. This is because after passing through a secret state, the status of indicating may become non-indicating.*

Therefore, if the system is at a state whose K-step reachable set is a subset of indicator state, then based on this state information, one can predict K-step ahead that a secret state will be visited. We define

$$\mathcal{N}_K := \{x \in X : R_K(x) \cap \mathcal{N} \neq \emptyset\} \subseteq X$$

as the set of states the intruder cannot make such a prediction. Then we have the following theorem.

Theorem 3. *System G is K-step trajectory pre-opaque if and only if*

$$\forall \alpha \in P(\mathcal{L}(G)) : \hat{\mathcal{E}}(\alpha) \cap \mathcal{N}_K \neq \emptyset.$$

Proof. ($\Rightarrow$) By contradiction. Suppose that G is K-step trajectory pre-opaque and assume that there exists a string $\alpha \in P(\mathcal{L}(G))$ such that $\hat{\mathcal{E}}(\alpha) \cap \mathcal{N}_K = \emptyset$ holds. According to Definition 3, for any $n \geq K$, there exists $x_0' \in X_0, s' \in \mathcal{L}_o(G, x_0'), s't_1t_2 \in \mathcal{L}(G, x_0')$ such that $P(s') = P(s) = \alpha, |t_1| = K$, $|t_2| = n - K$ and for any $w \in \overline{\{t_2\}}$, we have $f(x_0', s't_1w) \notin X_S$. Now, let us choose n such that $n \geq |X| + K$, i.e., $|t_2| \geq |X|$. Since $f(x_0', s't_1t_2)$ can pass through at most $|X|$ states, there are at least two repeated states that forms a cycle along the path of t_2 starting from $f(x_0', s't_1)$. This immediately implies that $f(x_0', s't_1) \in \mathcal{N}$. Furthermore, we have $f(x_0', s't_1) \in R_K(f(x_0', s'))$ since $|t_1| = K$. Therefore, we have $R_K(f(x_0', s')) \cap \mathcal{N} \neq \emptyset$, i.e., $f(x_0', s') \in \mathcal{N}_K$. Since $P(s') = \alpha$ and $s' \in E^*E_o \cup \{\epsilon\}$, we have $f(x_0', s') \in \hat{\mathcal{E}}(\alpha)$, which implies that $\hat{\mathcal{E}}(\alpha) \cap \mathcal{N}_K \neq \emptyset$. This, however, contradicts to our assumption.

($\Leftarrow$) By contradiction. Suppose that for any $\alpha \in P(\mathcal{L}(G))$, we have $\hat{\mathcal{E}}(\alpha) \cap \mathcal{N}_K \neq \emptyset$ and assume that G is not K-step trajectory pre-opaque, i.e., there exist a state $x_0 \in X_0$, a string $s \in \mathcal{L}_o(G, x_0)$ and an integer $n \geq K$ such that

$$\begin{aligned}&(\forall x_0' \in X_0)(\forall s' \in \mathcal{L}_o(G, x_0'), \forall t_1t_2 \in \mathcal{L}(G, f(x_0', s')) \text{ s.t.}\\ &[P(s) = P(s') \wedge |t_1| = K \wedge [|t_1t_2| = n]\\ &\Rightarrow [\exists w \in \overline{\{t_2\}} : f(x_0', s't_1w) \in X_S]\end{aligned} \tag{8}$$

Let us consider an arbitrary state x in $\hat{\mathcal{E}}(P(s))$. This means that there exist a state $x_0' \in X_0$ and a string $s' \in \mathcal{L}_o(G, x_0')$ such that $f(x_0', s') = x$ and $P(s') = P(s)$. However, according to Equation (8), any string of length n from state x must pass through a secret state between its Kth instant and its nth instant. This means that $R_K(x) \cap \mathcal{N} = \emptyset$, i.e., $x \notin \mathcal{N}_K$. Note that x is an arbitrary state in $\hat{\mathcal{E}}(P(s))$. Therefore, we have $\hat{\mathcal{E}}(P(s)) \cap \mathcal{N}_K = \emptyset$. However, this contradicts to our assumption that $\hat{\mathcal{E}}(\alpha) \cap \mathcal{N}_K \neq \emptyset$ for any $\alpha \in P(\mathcal{L}(G))$. □

Similar to the case instant pre-opacity, there also exists an upper bound for K in trajectory pre-opacity.

Proposition 5. *For any $K' > K \geq 2^{|X|} - 1$, system G is K'-step trajectory pre-opaque, if and only if, G is K-step trajectory pre-opaque.*

Proof. Still, it suffices to show that K'-step trajectory pre-opacity implies K-step trajectory pre-opacity. For the sake of contradiction, assume that G is not K-step trajectory pre-opaque but is $(K+1)$-step trajectory pre-opaque, where $K \geq 2^{|X|} - 1$. According to Theorem 3, we know that

- $\forall \alpha \in P(\mathcal{L}(G)) : \hat{\mathcal{E}}(\alpha) \cap \mathcal{N}_{K+1} \neq \emptyset$; and
- $\exists \beta \in P(\mathcal{L}(G)) : \hat{\mathcal{E}}(\beta) \cap \mathcal{N}_K = \emptyset$.

The above two conditions further imply that $R_K(\hat{\mathcal{E}}(\beta)) \cap \mathcal{N} = \emptyset$ and $R_{K+1}(\hat{\mathcal{E}}(\beta)) \cap \mathcal{N} \neq \emptyset$. In addition, since for any $i \in \mathbb{N}$, $R_i(\hat{\mathcal{E}}(\beta))$ has at most $|X|$ elements and at least one element, there are $(2^{|X|} - 1)$ choices for $R_i(\hat{\mathcal{E}}(\beta))$. Note that the cardinality of multi-set $\{R_j(\hat{\mathcal{E}}(\beta)) : j = 0, 1, \cdots, K\}$ is $K + 1 \geq 2^{|X|} > 2^{|X|} - 1$, which means that there exist two integers $0 \leq m < n \leq K$, such that $R_m(\hat{\mathcal{E}}(\beta)) = R_n(\hat{\mathcal{E}}(\beta))$. Then $R_{K+1+m-n}(\hat{\mathcal{E}}(\beta)) \cap \mathcal{N} = R_{K+1}(\hat{\mathcal{E}}(\beta)) \cap \mathcal{N} \neq \emptyset$. Assume that $x \in R_{K+1+m-n}(\hat{\mathcal{E}}(\beta)) \cap \mathcal{N}$, according to the definition of $\mathcal{N}$, we have that $R_{n-m-1}(\{x\}) \cap \mathcal{N} \neq \emptyset$. Since $R_{n-m-1}(\{x\}) \subseteq R_{n-m-1}(R_{K+1+m-n}(\hat{\mathcal{E}}(\beta)))$, we know that $R_{n-m-1}(R_{K+1+m-n}(\hat{\mathcal{E}}(\beta))) \cap \mathcal{N} \neq \emptyset$. Therefore,

$$R_K(\hat{\mathcal{E}}(\beta)) \cap \mathcal{N} = R_{n-m-1}(R_{K+1+m-n}(\hat{\mathcal{E}}(\beta))) \cap \mathcal{N} \neq \emptyset,$$

which forms a contradiction. This ends the proof. □

C. Verification Algorithms

Now, let us discuss how to use the derived necessary and sufficient conditions to verify K-step instant or trajectory pre-opacity. To this end, we need to compute

- All possible state estimates, i.e., $\{\hat{\mathcal{E}}(\alpha) : \alpha \in P(\mathcal{L}(G))\}$;
- A set of n-step indicator states for $K \leq n \leq K + 2^{|X|} - 1$, i.e., $\{\Im_K, \ldots, \Im_{K+2^{|X|}-1}\}$ (for instant pre-opacity);
- The set of states whose K-step reachable set contains at least a non-indicator state, i.e., $\mathcal{N}_K$ (for trajectory pre-opacity).

1) Computation of $\hat{\mathcal{E}}(\alpha)$: Note that, compared with the standard current-state estimate, the state estimate considered here does not contain the unobservable tail. This can be computed by a slightly modified version of the standard observer automaton (we still call it observer here for the sake of simplicity). Formally, the observer of G is a new FSA

$$Obs(G) = (Q_{obs}, E_o, f_{obs}, q_{obs,0}),$$

where $Q_{obs} \subseteq 2^X \backslash \emptyset$ is the set of states, E_o is the set of events, $q_{obs,0} = X_0$ is the initial state, and $f_{obs} : Q_{obs} \times E_o \to Q_{obs}$ is the deterministic transition function defined by: for any $q \in Q_{obs}, \sigma \in E_o$, we have

$$f_{obs}(q, \sigma) = \{x \in X : \exists x' \in q, w \in E_{uo}^* \text{ s.t. } f(x', w\sigma) = x\} \tag{9}$$

For the sake of simplicity, we only consider the reachable part of the observer. Then we have

$$\forall \alpha \in P(\mathcal{L}(G)) : f(q_{obs,0}, \alpha) = \hat{\mathcal{E}}(\alpha).$$

Therefore, all possible state estimate $\hat{\mathcal{E}}(\alpha)$ can be computed with complexity $O(|E_o|2^{|X|})$.

2) Computation of $\Im_n$: For any give $n \geq 0$, one can compute $\Im_n$ by backtracking n steps from the set of all secret states. Formally, one can define an operator $F : 2^X \to 2^X$ by: for any $q \in 2^X$, we have

$$F(q) = \{x \in X : \forall \sigma \in E, f(x, \sigma)! \text{ s.t. } f(x, \sigma) \in q\}. \tag{10}$$

Then one can easily check that

$$\Im_n = F^n(\Im_0) \text{ with } \Im_0 = X_S$$

which can be computed with complexity $O(n|E_o||X|)$.

3) Computation of $\mathcal{N}_K$: To compute $\mathcal{N}_K$, first we need to compute the set of non-indicator states $\mathcal{N}$. To this end, we can remove all secret states in G and compute all strongly connected components, i.e., cycles; this can be done by, e.g., Kosaraju's algorithm with a linear complexity in the size of G [47]. Then those states that can reach a non-secret cycle are the set of non-indicator states. Therefore, computing set $\mathcal{N}$ can be done in $O(|E||X|)$. In order to compute $\mathcal{N}_K$, one can backtrack from $\mathcal{N}$ using another operator $W : 2^X \to 2^X$ defined by: for any $q \in 2^X$, we have

$$W(q) = \{x \in X : \exists \sigma \in E \text{ s.t. } f(x, \sigma) \in q\}. \tag{11}$$

Then one can easily check that

$$\mathcal{N}_K = W^K(\mathcal{N}_0) \text{ with } \mathcal{N}_0 = \mathcal{N}$$

which can be computed with complexity $O(K|E_o||X|)$. Therefore, the overall complexity for computing set $\mathcal{N}_K$ is $O(K|E_o||X|)$.

Based on the above discussions, we summarize the algorithms for the verification of K-step instant pre-opacity and K-step trajectory pre-opacity by Algorithm INS-PRE-OPA-VER and Algorithm TRAJ-PRE-OPA-VER, respectively. The complexity of Algorithm INS-PRE-OPA-VER is $O(|E_o|2^{|X|}[K + (K+1) + \cdots + (K + 2^{|X|} - 1)]|E_o||X|) = O(|E_o|^2|X|(K + 2^{|X|-1})2^{2|X|})$ for the general case and is $O(K|E_o|^2|X|2^{|X|})$ under the assumption that there is no unobservable event. The complexity of Algorithm TRAJ-PRE-OPA-VER is simply $O(K|E_o|^2|X|2^{|X|})$, which is dominated by the size of the observer. We illustrate the verification algorithms by the following examples.

Algorithm 1: INS-PRE-OPA-VER

input : System G with X_S, E_o and K
output: YES or NO

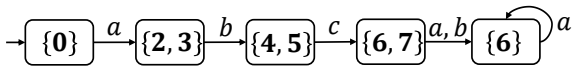
```
Construct the observer obs(G);
if there is no unobservable event in G then
    M ← 0;
else
    M ← 2^|X| − 1;
end
for q ∈ Q_obs do
    for n ∈ {K, K+1, ..., K+M} do
        if q ⊆ ℑ_n then
            return NO;
        end
    end
end
return YES;
```

Example 4. *Let us consider again system G_1 shown in Figure 2(a) and we verify whether or not it is K-step trajectory pre-opaque. First, we build its observer $Obs(G_1)$ as shown in Figure 6(a). The only non-indicator state is 6, i.e., $\mathcal{N} = \{6\}$. For $K = 2$, we have $\mathcal{N}_2 = W^2(\{6\}) = \{2, 4, 5, 6, 7\}$. Since $\{0\} \cap \{2, 4, 5, 6, 7\} = \emptyset$, we know that G_1 is not 2-step trajectory pre-opaque. However, for $K = 3$, we have*

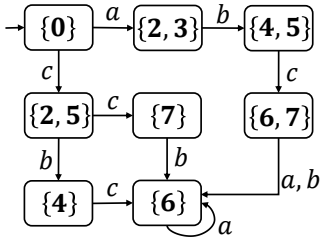
Algorithm 2: TRAJ-PRE-OPA-VER

input : System G with X_S, E_o and K
output: YES or NO

```
Construct the observer obs(G);
for q ∈ Q_obs do
    if q ∩ N_K = ∅ then
        return NO;
    end
end
return YES;
```



(a) $Obs(G_1)$



(b) $Obs(G_2)$

Fig. 6. Observers for G_1 and G_2, respectively.

$\mathcal{N}_3 = W^3(\{6\}) = \{0, 2, 3, 4, 5, 6, 7\}$ and each observer state has a common element with $\mathcal{N}_3$. Therefore, G_1 is 3-step trajectory pre-opaque. These results are also consistent with our previous intuitive analysis.

However, this system is K-step instant pre-opaque for any $K \geq 0$. To see this, it suffices to consider the case of $K = 0$. In this case, we have

$$\Im_0 = \{4, 7\}, \Im_1 = \{2, 5\}, \Im_2 = \{3\}, \Im_3 = \{1\},$$
$$\Im_4 = \Im_5 = \cdots = \emptyset$$

Therefore, no observer state is a subset of any $\Im_i$, which implies 0-step instant pre-opacity.

Example 5. *For system G_2 shown in Figure 2(b), its observer is shown in Figure 6(a). Then we have*

$$\Im_0 = \{4, 7\}, \Im_1 = \{2, 5\}, \Im_2 = \{3\}, \Im_3 = \Im_4 = \cdots = \emptyset$$

However, for observer state $\{2, 5\}$, we have $\{2, 5\} \subseteq \Im_1$, which means that G_2 is not 1-step instant pre-opaque. On the other hand, G_2 is K-step instant pre-opaque for any $K \geq 2$ as no state in $Obs(G_2)$ is a subset of any $\Im_n, n \geq 2$.

D. The Complexity of K-Step Pre-Opacity

Note that the complexity of Algorithm INS-PRE-OPA-VER and Algorithm TRAJ-PRE-OPA-VER are both exponential in

the number of states in G. Next, we show that both properties are essentially PSPACE-hard; therefore, the exponential complexity seems to be unavoidable.

Theorem 4. *Deciding whether or not G is K-step instant (or trajectory) pre-opaque is PSPACE-hard even when G is deterministic.*

Proof. Given two non-deterministic automata (NFAs) $G_1 = (X_1, E, f_1, X_{1,0})$ and $G_2 = (X_2, E, f_2, X_{2,0})$, the problem of language containment asks to decide whether or not $\mathcal{L}(G_1) \subseteq \mathcal{L}(G_2)$. This problem is known to be PSPACE-hard. Hereafter, we show that checking K-step instant/trajectory pre-opacity is also PSPACE-hard by reducing the language containment problem to the pre-opacity verification problem.

Let $G_1 = (X_1, E, f_1, X_{1,0})$ and $G_2 = (X_2, E, f_2, X_{2,0})$ be two NFAs with initial states $X_{1,0}$ and $X_{2,0}$, respectively. Without loss of generality, we assume G_1 and G_2 are live; otherwise, we can add a self-loop with a new event at each state in G_1 and G_2. Note that, in the analysis of pre-opacity, we assume that the transition function is deterministic; this gap can be bridged by using unobservable events to mimic non-determinism. Formally, let $E_u = \{u_1, u_2, \ldots, u_k\}$ be a set of new unobservable events. Then for each NFA G_i, we construct a new FSA $\tilde{G}_i = (\tilde{X}_i, \tilde{E}, \tilde{f}_i, \tilde{X}_{i,0})$ by: $\tilde{X}_i = X_i \cup \{(x,\sigma) \in X_i \times E : f_i(x,\sigma)!\}$, $\tilde{E} = E \cup E_u$, $X_{i,0} = \tilde{X}_{i,0}$, and $\tilde{f}_i : \tilde{X}_i \times \tilde{E} \to \tilde{X}_i$ is the deterministic transition function defined by: for any $f_i(x,\sigma)!$, we have $\tilde{f}_i(x,\sigma) = (x,\sigma)$ and $f_i(x,\sigma) = \{\tilde{f}_i((x,\sigma),u) : u \in E_u\}$. The construction of $\tilde{G}_i$ is illustrated by Figure 7. Clearly, one has $\mathcal{L}(G_1) \subseteq \mathcal{L}(G_2)$ iff $P(\mathcal{L}(\tilde{G}_1)) \subseteq P(\mathcal{L}(\tilde{G}_2))$.

Now we construct a new FSA $\tilde{G} = (\tilde{X}, \tilde{E}, \tilde{f}, \tilde{X}_0)$ by taking the union of $\tilde{G}_1$ and $\tilde{G}_2$, i.e., $\tilde{X} = \tilde{X}_1 \cup \tilde{X}_2$, $\tilde{f}$ is consistent with $\tilde{f}_1$ and $\tilde{f}_2$, and $\tilde{X}_0 = \tilde{X}_{1,0} \cup \tilde{X}_{2,0}$. Then, for system $\tilde{G}$, we let $X_S = X_1$ and E_u be the set of unobservable events. We show that $\tilde{G}$ is 0-step instant (or trajectory) pre-opaque if and only if $\mathcal{L}(G_1) \subseteq \mathcal{L}(G_2)$.

($\Rightarrow$) To see this, we suppose that $\mathcal{L}(G_1) \not\subseteq \mathcal{L}(G_2)$, then we know that there exists a string $s \in \mathcal{L}(G_1) \setminus \mathcal{L}(G_2)$, i.e., there exists a string $t \in P(\mathcal{L}(\tilde{G}_1)) \setminus P(\mathcal{L}(\tilde{G}_2))$, since $\mathcal{L}(G_1) \subseteq \mathcal{L}(G_2)$ is equivalent to $P(\mathcal{L}(\tilde{G}_1)) \subseteq P(\mathcal{L}(\tilde{G}_2))$. Therefore, after observing t in $\tilde{G}$, since $X_S = X_1$, we know for sure that the system now is at a secret state and will be at secret states for any future instant. Hence, $\tilde{G}$ is not 0-step instant (or trajectory) pre-opaque.

($\Leftarrow$) Suppose that $\mathcal{L}(G_1) \subseteq \mathcal{L}(G_2)$ and we assume that, for the sake of contradiction, $\tilde{G}$ is not 0-step trajectory pre-opaque, which means it is also not 0-step instant pre-opaque. Then we know that there exists a string $s \in P(\mathcal{L}(\tilde{G}))$ such that $\hat{\mathcal{E}}(s) \cap \mathcal{N}_0 = \emptyset$. Note that we have $P(\mathcal{L}(\tilde{G}_1)) \subseteq P(\mathcal{L}(\tilde{G}_2))$. Since $\mathcal{L}(G_1) \subseteq \mathcal{L}(G_2)$, this also implies that $P(\mathcal{L}(\tilde{G})) = P(\mathcal{L}(\tilde{G}_2))$ and $s \in P(\mathcal{L}(\tilde{G}_2))$. However, since every state in $\tilde{G}_2$ is non-secret and $\tilde{G}_2$ is live, we have $\tilde{X}_2 \subseteq \mathcal{N}_0$. Therefore, it is not possible that $\hat{\mathcal{E}}(s) \cap \mathcal{N}_0 = \emptyset$, which is a contradiction.

Overall, we conclude that deciding whether or not G is K-step instant/trajectory pre-opacity is PSPACE-hard. □

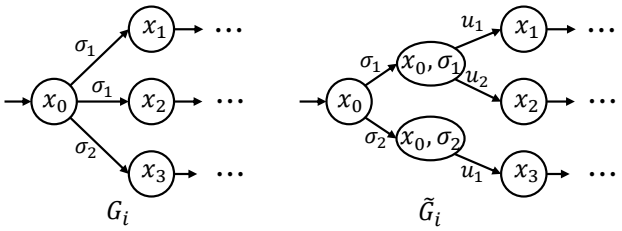


Fig. 7. Conceptual illustration of how to construct $\tilde{G}_i$ from G_i

.

V. Secret Intention as a Sequence Pattern

In the previous sections, the secret intention of the system is interpreted as reaching some secret states. In this section, we further generalize this setting by considering the secret intention as the willing to execute some particular sequences of events, which we call a *sequence pattern*. We present an illustrative example that motivates the definition of pattern pre-opacity and show how it can be reduced to state-based pre-opacity.

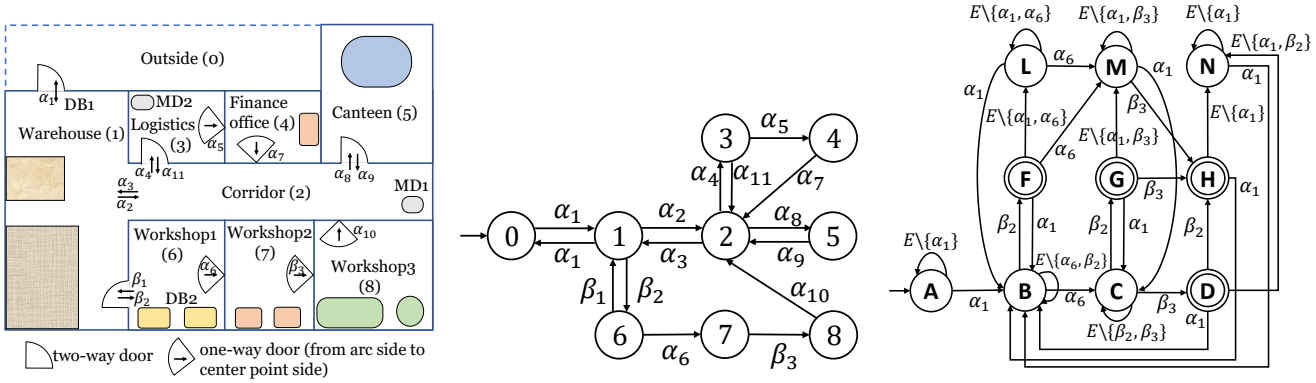
A. Illustrative Example of Pattern Pre-Opacity

We consider a location-tracking/prediction type problem in a smart factory building equipped with sensors as shown in Figure 8(a). The factory has eight regions of interest: a *warehouse*, a *logistics*, a *finance office*, a *canteen*, a *corridor* and three *workshops*. We assume there is a person in the factory that can move from one region to another by passing through a door; some doors are one-way and some are two-way as depicted in the figure. In particular, there are two doors DB_1 and DB_2 secured by door barrier sensors, which allow to observe if a person crosses the door but cannot tell the direction. Furthermore, there are two additional motion detector sensors (MD_1 and MD_2) at corridor and logistics, respectively; they can detect if a person moves to the corridor (or logistics) and specify the direction of the movement. The building monitor is able to use these sensors to track and predict the behavior of the person.

According to the structure of factory and different types of doors, the overall system, which is the mobility of the person, can be modeled as a DES as shown in Figure 8(b), where states 0 to 8 represent, respectively, regions *outside*, *warehouse*, *corridor*, *logistics*, *finance office*, *canteen*, *workshop 1*, *workshop 2* and *workshop 3*. Based on the distribution of motion detector sensors and door barrier sensors, we know that the set of observable events is

$$E_o = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7, \alpha_8, \alpha_9, \alpha_{10}\}.$$

Now we assume that the person wants to move in the factory to complete two tasks "secretly": (*task 1*) first goes to *warehouse* and then goes to *workshop 1*; (*task 2*) first goes to *warehouse*, and then enters *workshop 2*, and finally gets to *workshop 3*. Furthermore, the person wants to hide its intention for executing the above sequences against the monitor before they are completed. In this setting, "secret intention" is no longer visiting a secret state in the future. Instead, completing



(a) The topology of a factory (b) The system automaton G_3 (c) G_Ω

Fig. 8. An illustrative case of pattern pre-opacity.

any sequence containing string $\alpha_1\beta_2$ or $\alpha_1\alpha_6\beta_3$ can be viewed as a secret behavior. One can check that the person may not be able to hide its intention to complete task 2 more than one step before its completion. This is because once motion detector sensor DB_2 is triggered, the building monitor can determine for sure that the person was from *warehouse*, and is currently at *workshop 2* and will go to *workshop 3* in one step to complete the task. To formally describe this scenario, we propose K-step instant/trajectory pattern pre-opacity in the next part.

B. Definitions of Pattern Pre-Opacity

Now, we formally formulate the notion of pattern pre-opacity. Specifically, we consider a regular *pattern language* $\Omega \subseteq E^*$ which is the set of all strings containing the secret sequence patterns of the system. Then we say that a system is K-step pattern pre-opaque if any completion of a string in the pattern language can be predicted K-step ahead. Depending on whether or not the intruder needs to determine the specific instant of the completion, pattern pre-opacity can also be categorized as instant pre-opacity and trajectory pre-opacity.

Definition 4. *(K-Step Instant Pattern Pre-Opacity) Given system G, set of observable events E_o, pattern language Ω, and non-negative integer $K \in \mathbb{N}$, system G is said to be K-step instant pattern pre-opaque (w.r.t. E_o and Ω) if*

$$\begin{aligned}&(\forall x_0 \in X_0)(\forall s \in \mathcal{L}_o(G, x_0))(\forall n \geq K)\\&(\exists x_0' \in X_0)(\exists s' \in \mathcal{L}_o(G, x_0'), t \in \mathcal{L}(G, f(x_0', s'))\\&[P(s) = P(s') \wedge |t| = n \wedge s't \notin \Omega]\end{aligned} \tag{12}$$

Definition 5. *(K-Step Trajectory Pattern Pre-Opacity) Given system G, set of observable events E_o, a pattern language Ω, and non-negative integer $K \in \mathbb{N}$, system G is said to be K-step trajectory pattern pre-opaque (w.r.t. E_o and Ω) if*

$$\begin{aligned}&(\forall x_0 \in X_0, \forall s \in \mathcal{L}_o(G, x_0))(\forall n \geq K)\\&(\exists x_0' \in X_0, \exists s' \in \mathcal{L}_o(G, x_0'), \exists t_1 t_2 \in \mathcal{L}(G, f(x_0', s')) \text{ s.t.}\\&[P(s) = P(s')] \wedge [|t_1| = K] \wedge [|t_1 t_2| = n] \wedge\\&\quad [\forall w \in \overline{\{t_2\}} : s' t_1 w \notin \Omega]\end{aligned}$$

Intuitively, K-step trajectory pattern pre-opacity says that, for any observation, the intruder cannot predict K-step ahead that a secret sequence pattern will be completed. The definition of instant pattern pre-opacity is similar; the only difference is that it also requires to specify the specific instant of the completion. Clearly, pre-opacity is a special case of pattern pre-opacity as we can define all sequences reaching secret states as the sequence pattern. Hereafter, we will show that pattern pre-opacity can also be transformed to standard pre-opacity by refining the state-space and suitably defining secret states.

Example 6. *Consider again the example shown in Figure 8(b). The pattern language can be described by*

$$\begin{aligned}\Omega =&((E\backslash\{\alpha_1\})^*\{\alpha_1\}(E\backslash\{\alpha_6\})^*\{\alpha_6\}(E\backslash\{\beta_3\})^*\{\beta_3\}\\&\cup (E\backslash\{\alpha_1\})^*\{\alpha_1\}(E\backslash\{\beta_2\})^*\{\beta_2\})^*\end{aligned} \tag{13}$$

Essentially, regular pattern language Ω includes all strings that contain sequence pattern $\alpha_1\alpha_6\beta_3$ or $\alpha_1\beta_2$. This language can be marked by FSA G_Ω shown in Figure 8(c). Obviously, G_3 is 2-step instant pattern pre-opaque, since based on any observation, the intruder cannot know for sure the system will finish a sequence pattern $\alpha_1\beta_2$ or $\alpha_1\alpha_6\beta_3$ 2-step ahead. However, as we discussed early, it is not 1-step instant pattern pre-opaque; this is because, once string $\alpha_1\alpha_6$ is observed, the monitor knows for sure that sequence pattern $\alpha_1\alpha_6\beta_3$ will be completed in 1-step. Also, we can check that G_3 is 2-step trajectory pattern pre-opaque but not 1-step trajectory pattern pre-opaque.

Note that when string $\alpha_1\alpha_6$ is observed, we know that sequence pattern $\alpha_1\beta_2$ has been finished one step ago. Although the monitor fails to detect sequence pattern $\alpha_1\beta_2$ before its completion, it still can predict sequence pattern $\alpha_1\alpha_6\beta_3$.

Remark 5. *The concept of* sequence pattern *was first proposed in the literature for the purpose of fault diagnosis [48] and fault prognosis [41]. Specifically, a sequence pattern is used to model the set of behaviors considered as fault. Our notion of sequence pattern is more general than that in the context of fault diagnosis/prognosis. In particular, in the context of fault diagnosis/prognosis, the sequence pattern is*

assumed to be stable *in the sense that any continuation of a sequence in the pattern is still in the pattern. This is motivated by the setting of permanent fault. However, our definition of sequence pattern does not necessarily be stable as the system can be secret/non-secret intermittently. In other words, even if the intruder miss the predication of the first sequence pattern, it may still be able to predict some future sequence pattern, and in this case, the system is also not pre-opaque.*

C. Verifications of Pattern Pre-Opacity

We show how to verify pattern pre-opacity in this part. To this end, we assume that the secret pattern language Ω is a regular language and it is recognized by a FSA $G_\Omega = (X_\Omega, E, f_\Omega, x_{0,\Omega}, X_{m,\Omega})$, i.e., $\mathcal{L}_m(G_\Omega) = \Omega$, where $x_{0,\Omega}$ is the unique initial state. Without loss of generality, we assume that G_Ω is total, i.e., $\mathcal{L}(G_\Omega) = E^*$; otherwise, we can add a new unmarked "dump" state and complete the transition function.

Then let $G = (X, E, f, X_0)$ be the system and $G_\Omega = (X_\Omega, E, f_\Omega, x_{0,\Omega}, X_{m,\Omega})$ be the FSA recognizing the sequence pattern. We define the product of G and G_Ω as

$$G_\times = (X', E', f', X'_0),$$

where $X' \subseteq X \times X_\Omega, E' = E, X'_0 = X_0 \times \{x_{0,\Omega}\}$ and $f' : X' \times E \to X'$ is the transition function defined by $f((x_1, x_2), \sigma) = (f(x_1, \sigma), f_\Omega(x_2, \sigma))$, if $f(x_1, \sigma)$ and $f_\Omega(x_2, \sigma)$ are defined, and undefined otherwise. Then we define

$$X'_S = \{(q_1, q_2) : q_2 \in X_{m,\Omega}\}$$

as the set of secret states in $G_\times$. Then the following result shows that pattern pre-opacity can be transformed to state-based pre-opacity.

Theorem 5. *System G is K-step instant (respectively, trajectory) pattern pre-opaque w.r.t. Ω if and only if $G \times G_\Omega$ is K-step instant (respectively, trajectory) pre-opaque w.r.t. X'_S.*

Proof. We only show the case of instant pre-opacity; the case of trajectory pre-opacity is similar.

($\Rightarrow$) Suppose that $G \times G_\Omega$ is not K-step instant pre-opaque, which implies that

$$\begin{aligned}
&(\exists (x_0, x_{0,\Omega}) \in X'_0)(\exists s \in \mathcal{L}_o(G \times G_\Omega, (x_0, x_{0,\Omega})))(\exists n_0 \geq K)\\
&(\forall (x'_0, x'_{0,\Omega}) \in X'_0)\\
&(\forall s' \in \mathcal{L}_o(G \times G_\Omega, (x'_0, x'_{0,\Omega})), s't \in \mathcal{L}(G \times G_\Omega, (x'_0, x'_{0,\Omega})))\\
&[P(s) = P(s') \wedge |t| = n_0] \Rightarrow [f'((x'_0, x'_{0,\Omega}), s't) \in X'_S]
\end{aligned}$$

Since G_Ω is complete, we have $\mathcal{L}(G) \subseteq \mathcal{L}(G_\Omega)$, then we know that for any $x'_0 \in X_0$, any $s' \in \mathcal{L}_o(G, x'_0), s't \in \mathcal{L}(G, x'_0)$ such that $P(s) = P(s')$ and $|t| = n_0 \geq K$, we have that $f_\Omega(x'_0, s't) \in X_{m,\Omega}$, i.e., $s't \in \mathcal{L}_m(G_\Omega) = \Omega$. This implies that G is not K-step instant pattern pre-opaque

($\Leftarrow$) Assume that G is not K-step instant pattern pre-opaque, i.e.,

$$\begin{aligned}
&(\exists x_0 \in X_0)(\exists s \in \mathcal{L}_o(G, x_0))(\exists n_0 \geq K)\\
&(\forall x'_0 \in X_0)(\forall s' \in \mathcal{L}_o(G, x'_0), s't \in \mathcal{L}(G, x'_0))\\
&[P(s) = P(s') \wedge |t| = n_0 \wedge s't \in \Omega]
\end{aligned}$$

Since $\mathcal{L}(G \times G_\Omega) \subseteq \mathcal{L}(G)$, we know that for any $(x'_0, x_{0,\Omega}) \in X'_0, s' \in \mathcal{L}_o(G \times G_\Omega, (x'_0, x_{0,\Omega}))$ and $s't \in \mathcal{L}(G \times G_\Omega, (x'_0, x_{0,\Omega}))$ such that $P(s') = P(s)$ and $|t| = n_0$, we always have $f'((x'_0, x'_{0,\Omega}), s't) \in X'_S$ and $|t| = n_0 \geq K$, which means that $G \times G_\Omega$ is not K-step instant pre-opaque. □

Example 7. *Consider again system automaton G_3 and pattern automaton G_Ω in Figure 8. To verify the K-step instant/trajectory pattern pre-opacity of G_3, we fist construct $G_3 \times G_\Omega$, which is omitted here for the sake of brevity. Then the set of secret states in $G_3 \times G_\Omega$ is $X'_S = \{(6, F), (8, H)\}$. One can verify that $G_3 \times G_\Omega$ is 2-step instant pre-opaque but not 1-step instant pre-opaque; also, $G_3 \times G_\Omega$ is 2-step trajectory pre-opaque but not 1-step trajectory pre-opaque. Therefore, based on Theorem 5, for sequence patterns captured by Ω, we know that G_3 is 2-step instant pattern pre-opaque but not 1-step instant pattern pre-opaque, and it is 2-step trajectory pattern pre-opaque but not 1-step trajectory pattern pre-opaque, which are consistent with our previous analysis.*

VI. Conclusion

In this paper, we proposed the notion of pre-opacity to verify the *intention security* of a partially-observed DES. Two notions of pre-opacity called K-step instant pre-opacity and K-step trajectory pre-opacity are proposed. For each notion of pre-opacity, we provide a verifiable necessary and sufficient condition as well as an effective verification algorithm. We also generalize the notions of pre-opacity to the case where the secret behavior is captured by a sequence pattern. Our work extends the theory of opacity to a new class where secret is related to the intention of the system. We believe there are many interesting future directions related to the concept of pre-opacity. One interesting direction is to *synthesize* a supervisor to enforce pre-opacity when the verification result is negative. Also, we would like to extend the notion of pre-opacity to the stochastic setting to quantitatively evaluate the information leakage.

References

[1] A. Saboori and C. N. Hadjicostis. Verification of k-step opacity and analysis of its complexity. *IEEE Transactions on Automation Science and Engineering*, 8(3):549–559, 2011.

[2] A. Saboori and C. N. Hadjicostis. Verification of initial-state opacity in security applications of discrete event systems. *Information Sciences*, 246:115–132, 2013.

[3] X. Yin and S. Lafortune. A new approach for the verification of infinite-step and K-step opacity using two-way observers. *Automatica*, 80:162–171, 2017.

[4] S. Mohajerani and S. Lafortune. Transforming opacity verification to nonblocking verification in modular systems. *IEEE Transactions on Automatic Control*, 65(4):1739–1746, 2019.

[5] J.W. Bryans, M. Koutny, L. Mazaré, and P. Ryan. Opacity generalised to transition systems. *Internationa Journal of Information Security*, 7(6):421–435, 2008.

[6] K. Zhang, X. Yin, and M. Zamani. Opacity of nondeterministic transition systems: A (bi) simulation relation approach. *IEEE Transactions on Automatic Control*, 64(12):5116–5123, 2019.

[7] J.W. Bryans, M. Koutny, and P. Ryan. Modelling opacity using Petri nets. *Electronic Notes in Theoretical Computer Science*, 121:101–115, 2005.

[8] Y. Tong, Z. Li, C. Seatzu, and A. Giua. Verification of state-based opacity using Petri nets. *IEEE Trans. Automatic Control*, 62(6):2823–2837, 2017.

[9] Y. Tong, Z. Li, C. Seatzu, and A. Giua. Decidability of opacity verification problems in labeled Petri net systems. *Automatica*, 80:48–53, 2017.

[10] X. Cong, M.P. Fanti, A.M. Mangini, and Z. Li. On-line verification of current-state opacity by petri nets and integer linear programming. *Automatica*, 94:205–213, 2018.

[11] B. Ramasubramanian, W.R. Cleaveland, and S. Marcus. Notions of centralized and decentralized opacity in linear systems. *IEEE Transactions on Automatic Control*, 265(4):1442–1455, 2020.

[12] L. An and G.-H. Yang. Opacity enforcement for confidential robust control in linear cyber-physical systems. *IEEE Transactions on Automatic Control*, 65(3):1234–1241, 2020.

[13] X. Yin, M. Zamani, and S. Liu. On approximate opacity of cyber-physical system. *IEEE Transactions on Automatic Control*, 2020.

[14] S. Takai and Y. Oka. A formula for the supremal controllable and opaque sublanguage arising in supervisory control. *SICE J. Control, Measu. & Syst. Integration*, 1(4):307–311, 2008.

[15] J. Dubreil, P. Darondeau, and H. Marchand. Supervisory control for opacity. *IEEE Trans. Automatic Control*, 55(5):1089–1100, 2010.

[16] F. Cassez, J. Dubreil, and H. Marchand. Synthesis of opaque systems with static and dynamic masks. *Formal Methods in System Design*, 40(1):88–115, 2012.

[17] P. Darondeau, H. Marchand, and L. Ricker. Enforcing opacity of regular predicates on modal transition systems. *Discrete Event Dyn. Sys.: Theory & Apl.*, 25(1-2):251–270, 2014.

[18] B. Zhang, S. Shu, and F. Lin. Maximum information release while ensuring opacity in discrete event systems. *IEEE Trans. Automation Science and Engineering*, 12(4):1067–1079, 2015.

[19] Y. Ji, Y.-C. Wu, and S. Lafortune. Enforcement of opacity by public and private insertion functions. *Automatica*, 93:369–378, 2018.

[20] B. Wu, J. Dai, and H. Lin. Synthesis of insertion functions to enforce decentralized and joint opacity properties of discrete-event systems. In *American Control Conference*, pages 3026–3031. IEEE, 2018.

[21] Y. Tong, Z. Li, C. Seatzu, and A. Giua. Current-state opacity enforcement in discrete event systems under incomparable observations. *Discrete Event Dynamic Systems*, 28(2):161–182, 2018.

[22] B. Behinaein, F. Lin, and K. Rudie. Optimal information release for mixed opacity in discrete-event systems. *IEEE Transactions on Automation Science and Engineering*, 16(4):1960–1970, 2019.

[23] Y. Ji, X. Yin, and S. Lafortune. Enforcing opacity by insertion functions under multiple energy constraints. *Automatica*, 108:108476, 2019.

[24] S. Mohajerani, Y. Ji, and S. Lafortune. Compositional and abstraction-based approach for synthesis of edit functions for opacity enforcement. *IEEE Transactions on Automatic Control*, 2020.

[25] A. Saboori and C.N. Hadjicostis. Coverage analysis of mobile agent trajectory via state-based opacity formulations. *Control Engineering Practice*, 19(9):967–977, 2011.

[26] Y.-C. Wu, K.A. Sankararaman, and S. Lafortune. Ensuring privacy in location-based services: An approach based on opacity enforcement. In *12th International Workshop on Discrete Event Systems*, pages 33–38, 2014.

[27] F. Lin, W. Chen, W. Wang, and F. Wang. Information control in networked discrete event systems and its application to battery management systems. *Discrete Event Dynamic Systems*, 30(2):243–268, 2020.

[28] A. Bourouis, K. Klai, N. Ben Hadj-Alouane, and Y. El Touati. On the verification of opacity in web services and their composition. *IEEE Transactions on Services Computing*, 10(1):66–79, 2017.

[29] R. Jacob, J.-J. Lesage, and J.-M. Faure. Opacity of discrete event systems: models, validation and quantification. *IFAC-PapersOnLine*, 48(7):174–181, 2015.

[30] S. Lafortune, F. Lin, and C.N. Hadjicostis. On the history of diagnosability and opacity in discrete event systems. *Annual Reviews in Control*, 45:257–266, 2018.

[31] F. Lin. Opacity of discrete event systems and its applications. *Automatica*, 47(3):496–503, 2011.

[32] Y.-C. Wu and S. Lafortune. Comparative analysis of related notions of opacity in centralized and coordinated architectures. *Discrete Event Dynamic Systems*, 23(3):307–339, 2013.

[35] A. Saboori and C.N. Hadjicostis. Current-state opacity formulations in probabilistic finite automata. *IEEE Transactions on Automatic Control*, 59(1):120–133, 2014.

[33] A. Saboori and C. N. Hadjicostis. Verification of infinite-step opacity and complexity considerations. *IEEE Transactions on Automatic Control*, 57(5):1265–1269, 2011.

[34] Y. Falcone and H. Marchand. Enforcement and validation (at runtime) of various notions of opacity. *Discrete Event Dynamic Systems*, 25(4):531–570, 2015.

[36] B. Bérard, K. Chatterjee, and N. Sznajder. Probabilistic opacity for Markov decision processes. *Information Processing Letters*, 115(1):52–59, 2015.

[37] C. Keroglou and C.N. Hadjicostis. Probabilistic system opacity in discrete event systems. *Discrete Event Dynamic Systems*, pages 1–26, 2017.

[38] J. Chen, M. Ibrahim, and R. Kumar. Quantification of secrecy in partially observed stochastic discrete event systems. *IEEE Trans. Automation Science and Engineering*, 14(1):185–195, 2017.

[39] B. Wu and H. Lin. Privacy verification and enforcement via belief abstraction. *IEEE Control Sys. Letters*, 2(4):815–820, 2018.

[40] X. Yin, Z. Li, W. Wang, and S. Li. Infinite-step opacity and K-step opacity of stochastic discrete-event systems. *Automatica*, 99:266–274, 2019.

[41] T. Jéron, H. Marchand, S. Genc, and S. Lafortune. Predictability of sequence patterns in discrete event systems. In *Proc. 17th IFAC World Congress*, pages 537–543, 2008.

[42] S. Genc and S. Lafortune. Predictability of event occurrences in partially-observed discrete-event systems. *Automatica*, 45(2):301–311, 2009.

[43] R. Kumar and S. Takai. Decentralized prognosis of failures in discrete event systems. *IEEE Trans. Autom. Contr.*, 55(1):48–59, 2010.

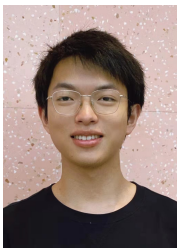
[44] S. Takai. Robust prognosability for a set of partially observed discrete event systems. *Automatica*, 51:123–130, 2015.

[45] J. Chen and R. Kumar. Stochastic failure prognosability of discrete event systems. *IEEE Trans. Autom. Contr.*, 60(6):1570–1581, 2015.

[46] X. Yin and Z.-J. Li. Decentralized fault prognosis of discrete event systems with guaranteed performance bound. *Automatica*, 69:375–379, 2016.

[47] R. Sedgewick and K. Wayne. *Algorithms*. Addison-Wesley Professional, 4th edition, 2011.

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



Shuo Yang was born in Hunan, China, in 2000. He is pursuing Ph.D degree at the Department of Electrical and Systems Engineering, University of Pennsylvania. He received the B.Eng degree in Automation from Shanghai Jiao Tong University in 2021. His current research interests include formal methods, robotics, and control.

Xiang Yin (S'14-M'17) was born in Anhui, China, in 1991. He received the B.Eng degree from Zhejiang University in 2012, the M.S. degree from the University of Michigan, Ann Arbor, in 2013, and the Ph.D degree from the University of Michigan, Ann Arbor, in 2017, all in electrical engineering. Since 2017, he has been with the Department of Automation, Shanghai Jiao Tong University, where he is an Associate Professor. His research interests include formal methods, discrete-event systems and cyber-physical systems.

Dr. Yin is serving as the co-chair of the *IEEE CSS Technical Committee on Discrete Event Systems*, an Associate Editor for the *Journal of Discrete Event Dynamic Systems: Theory & Applications*, and a member of the *IEEE CSS Conference Editorial Board*. Dr. Yin received the IEEE Conference on Decision and Control (CDC) Best Student Paper Award Finalist in 2016.