Verification of K-Step Opacity and Analysis of its Complexity

Anooshiravan Saboori and Christoforos N. Hadjicostis

***Abstract*—Motivated by security and privacy considerations in a variety of applications of discrete event systems, we describe and analyze the computational complexity required for verifying the notion of K-*step opacity* for systems that are modeled as non-deterministic finite automata with partial observation on their transitions. Specifically, a system is K-step opaque if, *at any specific point within the last* K *observations,* the entrance of the system state to a given set of *secret states* remains opaque (uncertain) to an intruder who has complete knowledge of the system model and observes system activity through some natural projection map. We provide two methods for verifying K-step opacity using two different state estimator constructions, and analyze the computational complexity of both.**

I. Introduction

The increased reliance of many applications on shared cyber-infrastructures (ranging from defense and banking to health care and power distribution systems) has led to various notions of *security and privacy*. A number of such notions focus on characterizing the *information flow* from the system to the intruder [1], [2]. *Opacity* falls in this category and aims at determining whether a given system's *secret* behavior (i.e., a subset of the behavior of the system that is considered critical and is usually represented by a predicate) is kept opaque to outsiders [3], [4]. More specifically, this requires that the intruder (modeled as a passive observer of the system's behavior) never be able to establish the truth of the predicate.

This material is based upon work supported in part by the U.S. National Science Foundation under NSF CNS Award 0834409. The research leading to these results has also received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements INFSO-ICT-223844 and PIRG02-GA-2007-224877. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of NSF or EC.

A. Saboori is with the Coordinated Science Laboratory, and the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign. C. N. Hadjicostis is with the Department of Electrical and Computer Engineering, University of Cyprus, and with the Coordinated Science Laboratory, and the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign. Corresponding author's address: 75 Kallipoleos Avenue, P.O. Box 20537, 1678 Nicosia, Cyprus. E-mail: chadjic@ucy.ac.cy, Tel: +357 22892231, Fax: +357 22892260.

In this paper, we consider opacity with respect to predicates that are state-based. More specifically, we consider a scenario where we are given a discrete event system (DES) that can be modeled as a non-deterministic finite automaton with partial observation on its transitions; assuming that the initial state of the system is (partially) known, we define the secret behavior of the system as the *evolution* of the system's state within a known subset of secret states S. Examples to motivate the study of such state-based notions of opacity are provided in our earlier work [5], and are briefly reviewed later in this paper for completeness. Among other applications, they include encryption using key strings provided by pseudo-random generators, coverage properties of mobile agents in sensor networks, and anonymity requirements in protocols for web transactions.

The paper defines and analyzes the state-based notion of K-*step opacity* (for $K \geq 0$) by requiring that the entrance of the system state to the set of secret states S, at any observation point within the past K observations, remain opaque to the intruder. In other words, in a K-step opaque system the intruder (which is assumed to have full knowledge of the system model and to be able to track the observable transitions in the system via the observation of associated labels) cannot determine with certainty that the state of the system $0, 1, \ldots,$ or K observations ago belonged to the set of secret states S. Our analysis starts by first establishing that a system is K-step opaque if and only if none of the k-delayed, $0 \leq k \leq K$, state estimates (i.e., estimates of the state of the system k observations ago which are consistent with all observations, including the last k observations) fall entirely within the set of secret states S. In order to capture the k-delayed state estimates ($0 \leq k \leq K$), we construct the K-*delay state estimator* (KDE) using two methods: (i) by storing the compatible K-delayed state estimate and remembering the last K observations, and (ii) by storing the compatible sequences of the last K-visited states. We compare the space complexity of the KDEs that result from these two methods and show that it is more state space efficient to store the compatible sequences of the last K-visited states than to store the sequence of the last K observations.

Apart from our own work in [4]–[6] (which looked at various state-based notions of opacity), the work in this paper is related to some of the existing security work in the area of DESs. In particular, [7] focuses on finite state Petri nets and defines opacity with respect to state-based predicates; our work in [4], [6] and in this paper essentially studies and solves this problem for the case of (non-deterministic) finite automata, also introducing in the process the notion of K-step opacity (not present in either [3] or [7]). The authors of [8] consider multiple intruders modeled as observers with different observation capabilities (namely different natural projection maps) and require that no intruder be able to determine that the actual trajectory of the system belongs to the secret language assigned to that intruder. Assuming that the supervisor can observe/control all events, sufficient conditions for the existence of a supervisor with a finite number of states are subsequently proposed. The assumptions on the controllability and observability of events are partially relaxed in [9] where the authors consider a single intruder that might observe different events than the ones observed/controlled by the supervisor. In contrast to [8] and [9] (which follow a language-based approach), opacity in our framework assumes that the states of the system can be partitioned into *secret* and *non-secret* ones; this state-based formulation is what leads to the use of various state estimators to verify opacity. Also, note that the notions of opacity introduced here are not considered in [8] and [9], and (as explained in [6]) they cannot be easily captured by the language framework of [8], [9] except for very special cases. Related to our work here is also the work in [10] where the authors partition the event set into high level and low level events, and consider the verification of *intransitive interference*, a property that captures the allowed information flow from high level events to low level events through a downgrading process (i.e., the inference of the occurrence of certain high level events from low level events). Our model of the intruder's capability (in terms of observability power) is different from [10] which makes the two frameworks hard to compare. However, for the case when there is no downgrading process, the notion of non-interference can be translated to an instance of 0-step opacity [6]. Note that, in general, one cannot formulate the notion of K-step opacity for $K > 0$ in the framework of [10].

II. Preliminaries and Background

Let Σ be an alphabet and denote by Σ^* the set of all finite-length strings of elements of Σ, including the empty string ϵ. For any string t, $|t|$ denotes the length of t (with $|\epsilon|$ taken to be zero). A language $L \subseteq \Sigma^*$ is a subset of finite-length strings in Σ^*. A language is finite if it contains only a finite number of strings. We say that a finite language L is of length K if the maximum length of the strings in L is K. For a string ω, $\overline{\omega}$ denotes the *prefix-closure* of ω and is defined as $\bar{\omega} = \{t \in \Sigma^* \mid \exists s \in \Sigma^* \{ts = \omega\}\}$ where ts denotes the concatenation of strings t and s. The post-string ω/t of ω after $t \in \bar{\omega}$ is defined as $\omega/t = s$ where $ts = \omega$ [11].

A DES is modeled in this paper as a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$, where $X = \{0, 1, \ldots, N-1\}$ is the set of states, Σ is the set of events, $\delta : X \times \Sigma \to 2^X$ (where 2^X is the power set of X) is the non-deterministic state transition function, and $X_0 \subseteq X$ is the set of possible initial states. The function δ can be extended from the domain $X \times \Sigma$ to the domain $X \times \Sigma^*$ in the routine recursive manner: $\delta(i, ts) := \bigcup_{j \in \delta(i,t)} \delta(j, s)$, for $t \in \Sigma$ and $s \in \Sigma^*$ with $\delta(i, \epsilon) := i$.The behavior of DES G is captured by $L(G) := \{s \in \Sigma^* \mid \exists i \in X_0 \{\delta(i, s) \neq \emptyset\}\}$. We use $L(G, i)$ to denote the set of all traces that originate from state i of G (so that $L(G) = \bigcup_{i \in X_0} L(G, i)$).

In general, only a subset Σ_{obs} of the events can be observed, so that Σ is partitioned into two sets, the set of observable events Σ_{obs} and the set of unobservable events Σ_{uo} (note that $\Sigma_{obs} \cap \Sigma_{uo} = \emptyset$ and $\Sigma_{obs} \cup \Sigma_{uo} = \Sigma$). The natural projection $P : \Sigma^* \to \Sigma_{obs}^*$ is typically used to map any trace executed in the system to the sequence of observations associated with it. This projection is defined recursively as $P(ts) = P(t)P(s),\ t \in \Sigma, s \in \Sigma^*$, with $P(t) = t$ if $t \in \Sigma_{obs}$ and $P(t) = \epsilon$ if $t \in \Sigma_{uo} \cup \{\epsilon\}$ [11]. [More general projections of the form $P : \Sigma \to L \cup \{\epsilon\}$ that may map multiple events to the same label from the set L can also be handled in a straightforward manner; to keep notation simple we only discuss the natural projection in this paper.]

Upon observing some string (sequence of observations) $\omega \in \Sigma_{obs}^*$, the state of the system might not be identifiable uniquely due to the lack of precise knowledge of the initial state, the non-determinism that is present in the state transition function, and the partial observation of events. We denote the set of states that the system is possibly in *given that ω was observed* as the (current) state estimate $\hat{X}_{|\omega|}(\omega)$. Similarly, we denote the set of states that the system was possibly in when it generated the K^{th} to last output (i.e., the state of the system K observations ago) following a sequence of observations $\omega = \alpha_0 \alpha_1 \ldots \alpha_n$ ($n \geq K$) as the K-delayed state estimate $\hat{X}_{|\omega|-K}(\omega)$ and define it formally bellow[1]. Note that the current-state estimate can also be

[1] K-delayed state estimation in discrete event systems is related to *fixed-lag smoothing* in discrete-time systems [12].

seen as the 0-delayed state estimate.

Definition 1 (K-Delayed State Estimate)**.** Given a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$ and a natural projection map P with respect to the set of observable events Σ_{obs} ($\Sigma_{obs} \subseteq \Sigma$), the K-delayed state estimate after observing string $\omega = \alpha_0\alpha_1 \dots \alpha_n$ ($n \geq K$) is defined as $\hat{X}_{|\omega|-K}(\omega) := \{j \in X \mid \exists t', t'' \in \Sigma^*, \exists i \in X_0\{j \in \delta(i, t'), \delta(j, t'') \neq \emptyset, P(t') = \alpha_0\alpha_1 \dots \alpha_{n-K}, P(t'') = \alpha_{n-K+1} \dots \alpha_n\}\}$. ■

Based on Definition 1, the K-delayed state estimate $\hat{X}_{|\omega|-K}(\omega)$ after observing $\omega = \alpha_0\alpha_1 \dots \alpha_n$ ($n \geq K$) is the set of all states that (i) are reachable in G from (at least one pair of) initial state i and a string t' with projection $P(t')$ equal to the first $n-K$ observable events in ω (in the same order) and (ii) for which there exists at least one continuation t'' with projection $P(t'')$ equal to the last K observable events in ω (in the same order). Note that the set of states reachable in G via a string t' with projection $P(t') = \alpha_0\alpha_1 \dots \alpha_{n-K} \equiv \omega'$ is the current state estimate that is obtained after observing ω' but before observing $P(t'') = \alpha_{n-K+1} \dots \alpha_n \equiv \omega''$; thus, $\hat{X}_{|\omega|-K}(\omega) \subseteq \hat{X}_{|\omega'|}(\omega')$ and the K-delayed state estimate can be seen as the subset of states in $\hat{X}_{|\omega'|}(\omega')$ from which the post K observations $\alpha_{n-K+1} \dots \alpha_n$ are possible. Note that Definition 1 implies that if $\omega \in \Sigma^*_{obs}$ is not a valid sequence of observations in G, then $\hat{X}_{|\omega|-K}(\omega) = \emptyset$. Also, by convention, $\hat{X}_{|\omega|-K}(\omega)$ is taken to be $\hat{X}_0(\omega)$ for $|\omega| < K$.

Given a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$, X^K ($K \geq 2$) denotes the set of K-tuples of states of DES G, i.e., $X^K := X \times X \times \dots \times X = \{(j_1, \dots, j_K) \mid j_k \in X, 1 \leq k \leq K\}$. We call $m \subseteq X^K$ a K-dimensional state mapping. Note that a 2-dimensional state mapping was called a *state mapping* in [5] and was introduced to analyze initial-state opacity.

The set of states included as the first (last) component in a K-dimensional state mapping m is called the set of starting (ending) states of m. We denote the set of starting states for K-dimensional state mapping m by $m(K-1)$ and the set of ending states by $m(0)$. We also denote by $m(k), 0 < k < K-1$, the set of intermediate states in the K-tuple, i.e.,

$$m(k) = \{j_{K-k} \mid (j_1, \dots, j_K) \in m, 0 \leq k \leq K-1\}.$$

We define the *shift* operator $>>: 2^{X^K} \times 2^{X^2} \to 2^{X^K}$ for a K-dimensional state mapping $m_1 \in 2^{X^K}$ and a state mapping $m_2 \in 2^{X^2}$ as

$$m_1 >> m_2 := \{(j_2, \dots, j_K, j_{K+1}) \mid (j_1, j_2, \dots, j_K) \in m_1, (j_K, j_{K+1}) \in m_2\}.$$

We also define the composition operator $\circ : 2^{X^2} \times 2^{X^2} \to 2^{X^2}$ for state mappings $m_1, m_2 \in 2^{X^2}$ as

$$m_1 \circ m_2 := \{(j_1, j_3) \mid \exists j_2 \in X\{(j_1, j_2) \in m_1, (j_2, j_3) \in m_2\}\}.$$

For any $Z \subseteq X$ and $K \geq 2$, we define the operator $\odot_K : 2^X \to 2^{X^K}$ as $\odot_K(Z) = \{(i, i, \dots, i) \mid i \in Z\}$ where the tuples involve K identical elements.

The 2-dimensional state mapping induced by a sequence of observations ω is defined as

$$M(\omega) = \{(i, j) \mid i, j \in X, \exists t \in \Sigma^*\{P(t) = \omega, j \in \delta(i, t)\}\}.$$

III. Problem Formulation

In this section, we formally define the notion of K-step opacity.

Definition 2 (K-Step Opacity)**.** Given a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$, a projection map P with respect to the set of observable events Σ_{obs} ($\Sigma_{obs} \subseteq \Sigma$), and a set of secret states $S \subseteq X$, automaton G is K-step opaque (for a nonnegative integer K) with respect to S and P (or (S, P, K)-opaque), if for all $t \in \Sigma^*$, $t' \in \bar{t}$, and $i \in X_0$,

$$\{|P(t)/P(t')| \leq K, \exists j \in S\{j \in \delta(i, t'), \delta(j, t/t') \neq \emptyset\}\}$$
$$\Rightarrow \{\exists s \in \Sigma^*, \exists s' \in \bar{s}, \exists i' \in X_0, \exists j' \in \delta(i', s')\{P(s) =$$
$$P(t), P(s') = P(t'), j' \in X - S, \delta(j', s/s') \neq \emptyset\}\}.$$ ■

Note that while the definition of K-step opacity studied in [6] is simpler and more intuitive, it is only suitable for deterministic automata; Definition 2, however, can be used for non-deterministic automata. For $t, s \in L(G)$ with $P(s) = P(t)$ we say that t passes through state j when s passes through state j' if there exists $t' \in \bar{t}$, $s' \in \bar{s}$, and $i, i' \in X_0$ such that $j \in \delta(i, t')$, $j' \in \delta(i', s')$ while (i) $P(t') = P(s')$ and (ii) t/t' and s/s' have continuations from states j and j', respectively. According to Definition 2, DES G is (S, P, K)-opaque if for every string t in $L(G)$ that visits a state j in S within the past K observations (and has a continuation from j), there exists a string s in $L(G)$ with $P(s) = P(t)$ such that when string t passes through the state j in S, string s passes through a state j' in $X-S$ (and has a continuation from j'). Note that s could be the same as t, in which case t would be passing through both secret and non-secret states.

Remark 3. The notion of K-step opacity is suitable for cases where there exists a bounded delay, after which one does not care if the outside observer can infer information about behavior that was previously considered secret (e.g., because the secret transaction has completed or because the intrusion will be detected). Motivated by applications where the existence of such

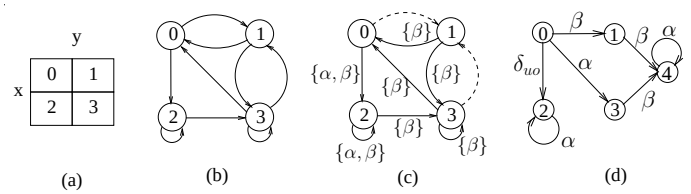

Fig. 1. (a) A 2-dimensional grid in which a vehicle can move; (b) Kinematic model H for a vehicle in the grid in (a); (c) Automaton G modeling the vehicle kinematic model and the corresponding sensor readings; (d) DES G in Section III-B.

bound might not be viable, we introduced in [13] the notion of *infinite-step opacity* which can be seen as (S, P, K)-opacity with $K \to \infty$ but requires different techniques for its verification. ■

A. Motivational Example

There are many areas where K-step opacity can be used to characterize security requirements of interest. In the sequel, we discuss an example in the context of tracking problems in sensor networks. More details can be found in [5].

Example 4. Consider a vehicle capable of moving on a two-dimensional space modeled as a 2-dimensional array of cells (in Figure 1-a we show a toy 2×2 grid). The vehicle possible movements in this space can be described via a kinematic model (a finite state machine) whose states are associated with the state (position) of the vehicle and whose transitions correspond to the possible movements of the vehicle at this position. Figure 1-b depicts an example of a kinematic model H for a vehicle that moves in the grid in Figure 1-a.

Typically, the sensor network that is deployed in this space will not capture all movements of the vehicle and hence the observation of movements will be partial. If each sensor detects the presence of the vehicle in a cell or in some aggregation of cells, then when the vehicle passes through a cell within the coverage of a sensor, this sensor emits a signal that indicates this event. Thus, we can enhance the kinematic model by assigning label α to all transitions that end in a cell that belongs to the coverage area of sensor α. Since sensor coverages may overlap, the label of transitions ending in areas which are covered by more than one sensor can be chosen to be a special label that indicates the set of all the sensors covering that location. In Figure 1-c, we depict the (non-deterministic) automaton G that models both the kinematic model of the vehicle and the corresponding sensor readings for a particular choice of sensor coverage areas. Dotted arrows correspond to transitions in locations that are not covered by any sensor.

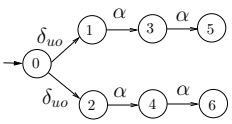

Fig. 2. DES G modeling a communication protocol for a bank transaction discussed in Section III-B.

One of the questions that might arise in the above context is that of characterizing all trajectories (sequences of states) that a vehicle can follow such that the passage of each trajectory from specific locations at specific observation points (points in time with respect to observation) remain ambiguous to the sensor network. These trajectories can be of interest for a variety of reasons. For example, they can be employed to hide the origin of a trajectory from an observer who is employing the sensor network (e.g., an observer who is observing the labels in Figure 1-c) trying to identify whether the origin belongs to a set of secret (strategically important) locations or whether the vehicle passed from this particular set of locations at some specific instant of time. Such questions can be answered using the opacity framework of this paper.

Note that a number of tools are already available for verifying notions of opacity, including K-step opacity (see for example [14]). ■

B. Related Notion: Trajectory-Based K-Step Opacity

It is easy to verify that the system in Figure 1-d is 2-step opaque with respect to $S = \{1, 6\}$; however, upon observing $\alpha\alpha$, the intruder is certain that, regardless of the state sequence that has occurred, the system has visited a secret state within the last 2 observations (although one cannot determine exactly when this happened). This system can be considered as insecure if the attacker is only interested in determining whether the system has reached secret states at any point during the last $K = 2$ observations. We refer to a system for which this scenario does not occur as a *trajectory-based K-step opaque* system. It is not hard to see that DES G is trajectory-based K-step opaque if and only if for any given sequence of observations ω, there always exists at least one sequence of states that G can follow such that only non-secret states are visited while generating the last K events in ω. Moreover, a system that is trajectory-based K-step opaque is also K-step opaque; but the converse is not necessarily true.

Note that the essential difference between K-step opacity and trajectory-based K-step opacity is the time at which the state of the system is exposed. Depending on the application, K-step opacity might be a more suitable requirement than trajectory-based K-step opacity

for characterizing security requirements. For instance, suppose the DES G in Figure III-B is a communication protocol for a bank transaction where a user has two options: communicate important account information while at state 1 (secret state) and dummy information while at states 3 and 5 (non-secret states), or communicate dummy information at states 2 and 4 (non-secret states) and important account information while at state 6 (secret state). If an eavesdropper does not know which of the two options the user has followed (due to the unobservable event δ_{uo}), then (even though she/he knows that important account information has been communicated) she/he does not know when this was done. Therefore, the fact that the system is not 2-step opaque is critical (despite the fact that the system is not trajectory-based 2-step opaque).

As another example, consider a pseudo-random generator that is used for generating a key string in encryption applications. Such a pseudo-random generator is usually implemented as an autonomous finite state machine that cycles through a large number of states. In this case, knowing that the system was in a particular state at a specific point in the past (as captured by K-step opacity) is indeed important because this exposes the subsequent sequence of states and thus the key string used for encryption. On the other hand, knowing that the system has been in a particular state in the recent past (as captured by trajectory-based K-step opacity) offers little information (in fact it offers zero information if K is larger than the number of states of the pseudo-random generator).

IV. Verification of K-Step Opacity using State Estimators

In this section, we show that for a DES G to be K-step opaque, it is necessary and sufficient that each k-delayed state estimate $\hat{X}_{|\omega|-k}(\omega)$, $0 \leq k \leq \min(K, |\omega|)$, associated with a sequence of observations ω contain at least one state outside the set of secret states S (unless the sequence of observations ω cannot be generated by G in which case $\hat{X}_{|\omega|-k}(\omega) = \emptyset$). The proof is straight forward and is not included due to space limitaitons. The reader can find it in [15].

Theorem 5. *Given a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$, a projection map P with respect to the set of observable events Σ_{obs} ($\Sigma_{obs} \subseteq \Sigma$), and a set of secret states $S \subseteq X$, automaton G is (S, P, K)-opaque if and only if for all $\omega \in \Sigma^*_{obs}$, $0 \leq k \leq \min(K, |\omega|)$*

$$\hat{X}_{|\omega|-k}(\omega) \not\subseteq S \text{ or } \hat{X}_{|\omega|-k}(\omega) = \emptyset, \qquad (1)$$

where $\hat{X}_{|\omega|-k}(\omega)$ is the k-delayed state estimate associated with the sequence of observations ω. ■

Existing state estimation techniques cannot verify K-step opacity since they are not tracking the k-delayed state estimates, $0 \leq k \leq K$. For this reason, in this paper we introduce the *K-delay state estimator* (KDE) which is a (deterministic) finite automaton that reconstructs the k-delayed state estimates ($0 \leq k \leq K$) associated with a given sequence of observations ω. In the sequel, we introduce two methods for constructing K-delay state estimators: (i) by storing the possible sequences of the last $(K+1)$-visited states via $(K+1)$-dimensional state mappings, (ii) by storing the k-delayed state estimates, $0 \leq k \leq K$, and remembering the sequence of the last K observations. We also discuss the state complexity of the KDEs that result from these two methods once we have the opportunity to describe them formally.

A. State Mapping-Based K-Delay State Estimator (SM-KDE)

The SM-KDE utilizes $(K+1)$-dimensional state mappings to capture the K-delayed state estimates as follows: each state of the SM-KDE is associated with a unique $(K+1)$-dimensional state mapping, with the initial state m_0 of the SM-KDE associated with the $(K+1)$-dimensional state mapping $\odot_{K+1}(X_0)$; with a slight abuse of notation we denote this by $m_0 = \odot_{K+1}(X_0)$. When observation $\alpha \in \Sigma_{obs}$ is made, this initial $(K+1)$-dimensional state mapping m_0 is shifted with the induced state mapping $M(\alpha)$ corresponding to observation α, resulting in a $(K+1)$-dimensional state mapping m_1 that associates with the next state of the state estimator, i.e., $m_1 = m_0 >> M(\alpha)$. Similarly, for each subsequent observation $\beta \in \Sigma_{obs}$, the current state of the SM-KDE that is associated with a $(K+1)$-dimensional state mapping m transitions into the state associated with the $(K+1)$-dimensional state mapping $m' = m >> M(\beta)$. From the structure of $(K+1)$-dimensional state mappings and the nature of the shift operator, we can establish that a sequence of observations causes the SM-KDE to transition through a sequence of $(K+1)$-dimensional state mappings to a $(K+1)$-dimensional state mapping m such that, at a given time step, the set of states in the state mapping m correspond to delayed state estimates. More specifically, the set of ending states $m(0)$ corresponds to zero-delayed state estimates, the set of intermediate states $m(k), 1 < k < K$, corresponds to k-delayed state estimates, and the set of starting states $m(K)$ corresponds to K-delayed state estimates. In this manner, we can build a structure which, at any time following a given sequence of observations, maintains information about the 0-delayed, 1-delayed, ..., and K-delayed state estimates through the $(K+1)$-dimensional state mappings associated with each of its

states. In fact, the estimator also contains complete information about the possible system state trajectories during the last K observations.

Definition 6 (State Mapping-Based K-Delay State Estimator (SM-KDE))**.** Given a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$ and a natural projection map P with respect to the set of observable events Σ_{obs} ($\Sigma_{obs} \subseteq \Sigma$), we define the K-delay state estimator as the deterministic automaton $G_{K,obs} = AC(2^{X^{(K+1)}}, \Sigma_{obs}, \delta_{K,obs}, X_{K,0})$ with state set $2^{X^{(K+1)}}$, event set Σ_{obs}, initial state $X_{K,0} = \odot_{K+1}(X_0)$, and state transition function $\delta_{K,obs} : 2^{X^{(K+1)}} \times \Sigma_{obs} \to 2^{X^{(K+1)}}$ defined for $\alpha \in \Sigma_{obs}$ as $m' = \delta_{K,obs}(m, \alpha) := m >> M(\alpha)$, where $m, m' \in 2^{X^{(K+1)}}$. [AC denotes the states of this automaton that are accessible starting from state $X_{K,0}$.] ■

Example 7. Consider the DES G in Figure 1-d with $X_0 = \{0, 1, 2, 3, 4\}$ and $\Sigma_{obs} = \{\alpha, \beta\}$. For this system, the 2-delay state estimator is represented in Figure 3-a along with the 3-dimensional state mappings $m_0, m_1, \ldots, m_{10}$ needed in the construction. The initial state of the system is $X_0 = X$ and the initial state of the 2-delay state estimator captures this in m_0 via a 3-dimensional state mapping that maps each system state to itself as $\odot_3(X) = \{(0,0,0), (1,1,1), (2,2,2), (3,3,3), (4,4,4)\} \equiv m_0$.

Starting from the initial state, assume that we observe α. The state mapping $M(\alpha)$ induced by observing α is $\{(0,2), (0,3), (2,2), (4,4)\}$ which implies that α can be observed only from states 0, 2 and 4. Moreover, if the initial state was 0, the current state can only be one of the states in $\{2, 3\}$; however, if the initial state was 2, the current state could only be 2; finally, if the initial state was 4, the current state would be 4. Following observation α, the next state m' in the 2-delay state estimator can be constructed as $m' = \delta_{K,obs}(m_0, \alpha) = m_0 >> M(\alpha) = \{(0,0,2), (0,0,3), (2,2,2), (4,4,4)\} \equiv m_1$.

Next, consider the case when following observation α we observe β. As the induced state mapping $M(\beta) = \{(0,1), (1,4), (3,4)\}$, we have $m' = \delta_{K,obs}(m_1, \beta) = m_1 >> M(\beta) = \{(0,3,4)\} \equiv m_4$. This implies that $\alpha\beta$ can only be observed if the system follows the state trajectory $0 \to 3 \to 4$. Using this approach for all possible observations (from each state), the 2-delay state estimator construction can be completed as shown in Figure 3-a. Note that we have not included the state that corresponds to the all empty state mapping (and any transitions from/to it) to avoid cluttering the diagram. ■

Remark 8. On the right of Figure 3-a, we use 3-dimensional trellis diagrams to describe the triples associated with states of the 2-delay state estimator. In general, we can graphically represent an induced K-dimensional state mapping m using a *K-dimensional trellis diagram*, i.e., a K-partite graph where the nodes in the state set X are replicated K times and ordered into K vertical slices ranging from slice 0 to slice $K-1$ (hence a K-dimensional trellis diagram has $K \cdot N$ nodes with $N = |X|$). Each node at slice k ($1 \leq k \leq K-2$) is either isolated or connected to (at least) one node at slice $k-1$ and (at least) one node at slice $k+1$. The nodes at slice 0 ($K-1$) are either isolated or connected to (at least) one node at slice 1 ($K-2$). ■

In the following theorem, we show that the SM-KDE state m that is reached via a sequence of observations ω is associated with a $(K+1)$-dimensional state mapping such that the first $|\omega| - K$ observations would have taken the system to the starting states of the $(K+1)$-dimensional state mapping and, in addition, the last K observations could have taken place from these starting states, visiting in the process the intermediate states in the tuple, in the order captured by the elements of the $(K+1)$-dimensional state mapping. The proof is provided in [15].

Theorem 9. *Suppose SM-KDE state m (as constructed in Definition 6) is reachable from the SM-KDE initial state $X_{K,0} = \odot_{K+1}(X_0)$ via the string $\omega = \alpha_0\alpha_1 \ldots \alpha_n$, $\omega \neq \epsilon$. Then, SM-KDE state m can be characterized as follows:*

(i) $|\omega| \leq K$*:* $m = \{(j_0, j_1, \ldots, j_K) \in X_0 \times X^K \mid (0 \leq l \leq |\omega| - 1, 0 \leq w \leq K - |\omega| - 1) : j_{w+1} = j_w, \exists t_l \in \Sigma^*\{P(t_l) = \alpha_l, j_{K-|\omega|+l+1} \in \delta(j_{K-|\omega|+l}, t_l)\}\}$.

(ii) $|\omega| \geq K$*:* $m = \{(j_0, j_1, \ldots, j_K) \in X^{K+1} \mid (0 \leq l \leq K-1) : \exists t_l \in \Sigma^*\{P(t_l) = \alpha_{n-K+1+l}, j_{l+1} \in \delta(j_l, t_l)\}, \exists i \in X_0, \exists t' \in \Sigma^*\{P(t') = \alpha_0\alpha_1 \ldots \alpha_{n-K}, j_0 \in \delta(i, t')\}\}$.

(iii) $m = \emptyset$ *when there is no* $t \in L(G)$ *such that* $P(t) = \omega$. ■

The proof of the following corollaries can also be found in [15].

Corollary 10. *The k-delayed state estimate $\hat{X}_{|\omega|-k}(\omega)$, $0 \leq k \leq \min(K, |\omega|)$, after observing $\omega = \alpha_0\alpha_1 \ldots \alpha_n$ is given by $\hat{X}_{|\omega|-k}(\omega) = m(k)$ where $m = \delta_{K,obs}(X_{K,0}, \omega)$.* ■

Corollary 10 proves that the SM-KDE captures the set of all k-delayed state estimates, $0 \leq k \leq K$, via its $(K+1)$-dimensional state mappings and, hence, by Theorem 5, it can be used for verifying K-step opacity.

Corollary 11. *Discrete event system G is (S, P, K)-opaque if and only if for all $m \in X_{K,obs}$, $k \in \{0, \ldots, K\}$,*

$$m(k) \not\subseteq S \text{ or } m(k) = \emptyset, \tag{2}$$

where $X_{K,obs}$ is the set of states in $G_{K,obs}$ that are reachable from the initial state $X_{K,0} = \odot_{K+1}(X_0)$. ■

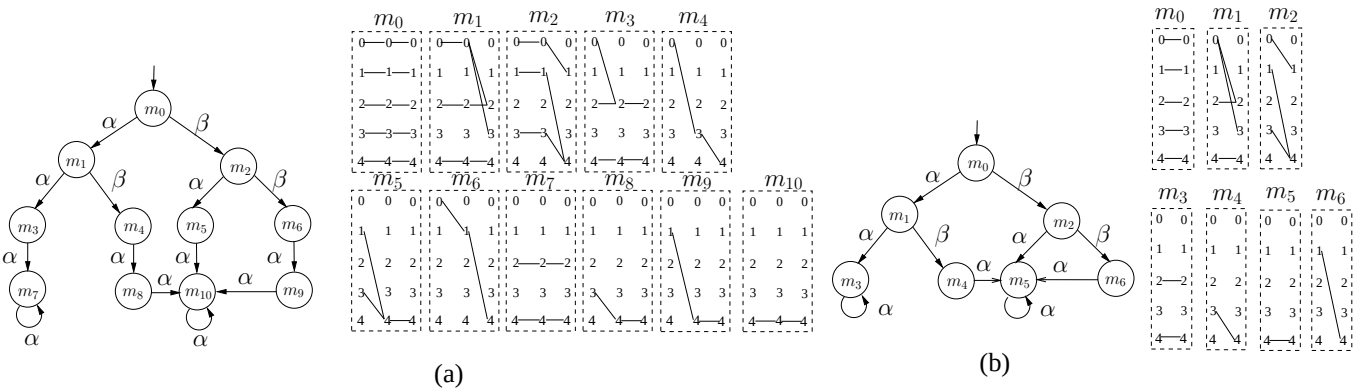

Fig. 3. (a) State mapping-based 2-delay state estimator corresponding to DES G; (b) State mapping-based 1-delay state estimator corresponding to DES G.

Example 12. DES G in Figure 1-d with $X_0 = \{0,1,2,3,4\}$ is not $(\{0\},P,2)$-opaque due to the existence of state m_4 (or m_6) in the state mapping-based 2-delay state estimator depicted in Figure 3-a. If the system generates the sequence of observations $\alpha\beta$ (or $\beta\beta$), then (since the only state from which $\alpha\beta$ or $\beta\beta$ can be observed is state 0) we can conclude with certainty that the system was in state 0 two steps ago. This violates the 2-step opacity requirement since state 0 is a secret state. The unit-delay state estimator for this system is shown in Figure 3-b (again we have not included the state that corresponds to the empty state mapping); it can be verified that for each of the 2-dimensional state mappings m associated with its states, every set of intermediate states $m(k), 0 \leq k \leq 1$, contains at least one element outside S. Hence, DES G is $(\{0\},P,1)$-opaque. ■

B. Observation Sequence-Based K-Delay State Estimator (OS-KDE)

In this section, we introduce the construction of automaton $G_{K,obs}^{observation}$ which captures K-delayed state estimates by remembering the sequence of the last K observations (this should be contrasted to $G_{K,obs}$ which captures the compatible sequences of the last K-visited states via $(K+1)$-dimensional state mappings). At each state of $G_{K,obs}^{observation}$, we store a $(K+2)$-tuple $Q \in \Sigma_{obs,\epsilon}^K \times 2^X \times \ldots \times 2^X$ consisting of the following information: *(i)* the last K observations ($\Sigma_{obs,\epsilon}$ denotes the set $\Sigma_{obs} \cup \{\epsilon\}$), and *(ii)* all the k-delayed state estimates for $k = 0,1,\ldots,K$. Upon observing a new event, the k-delayed state estimates are updated to ensure that estimates that are not consistent with the last observation are removed. Finally, the string that stores the last K observations is updated by adding the last observation to the end of it and by removing the first one. The main difference here is that $G_{K,obs}^{observation}$ only remembers the sets of states that are possible 0, 1, …, K observations ago but does not explicitly record the sequences of states that are possible; however, knowledge of the last K observations (together with the underlying system model) allows one to reconstruct these sequences if required. We now discuss the systematic construction of $G_{K,obs}^{observation}$. For brevity, we define the function $\delta_o : X \times \Sigma_{obs} \to 2^X$ for any $i \in X$ and $\alpha \in \Sigma_{obs}$ as

$$\delta_o(i,\alpha) = \{j \in X \mid \exists s \in \Sigma^* \{P(s) = \alpha, j \in \delta(i,s)\}\}. \tag{3}$$

The function δ_0 can be extended from the domain $X \times \Sigma_{obs}$ to the domain $X \times \Sigma_{obs}^*$ in the routine recursive manner: for $t \in \Sigma_{obs}$ and $s \in \Sigma_{obs}^*$, $\delta_o(i,ts) := \bigcup_{j \in \delta_o(i,t)} \delta_o(j,s)$, with $\delta_o(i,\epsilon) := i$. With a slight abuse of notation, we use $\delta_o : 2^X \times \Sigma_{obs} \to 2^X$ to also denote its extension from states to sets of states as follows: for all $Z \subseteq X$ define $\delta_o(Z,\alpha) = \bigcup_{z \in Z} \delta_o(z,\alpha)$. For clarity, we also introduce the following notation: let $\omega = \alpha'_0 \ldots \alpha'_n$ with $n \geq K$ denote the sequence of observations that have been seen so far (from the initialization of the system); we will rename the last K observations $\alpha'_{n-K+1} \ldots \alpha'_n$ to $\alpha_{-K+1} \ldots \alpha_0$ (i.e., $\alpha_{-i} = \alpha'_{n-i}$ for $i = 0,1,\ldots,K-1$) to make the discussion independent of n (the total number of observations seen so far). Also note that in the following definition strings $\alpha_{-k}\alpha_{-k+1}\ldots\alpha_0$ of length less than K are represented as $\epsilon^{K-k-1}\alpha_{-k}\alpha_{-k+1}\ldots\alpha_0$.

Definition 13 (Observation Sequence-Based K-Delay State Estimator (OS-KDE))**.** Given a non-deterministic finite automaton $G = (X,\Sigma,\delta,X_0)$ and a natural projection map P with respect to the set of observable events Σ_{obs} ($\Sigma_{obs} \subseteq \Sigma$), we define the observation sequence-based K-delay state estimator as the deterministic automaton $G_{K,obs}^{observation} = AC((\Sigma_{obs,\epsilon}^K \times 2^X \times \ldots \times 2^X), \Sigma_{obs}, \delta_{K,obs}^{observation}, X_{K,0}^{observation})$ with
(i) set of states $\Sigma_{obs,\epsilon}^K \times 2^X \times \ldots \times 2^X$, where each state is a $(K+2)$-tuple consisting of one string of length K or less, and $K+1$ subsets of X;
(ii) event set Σ_{obs};

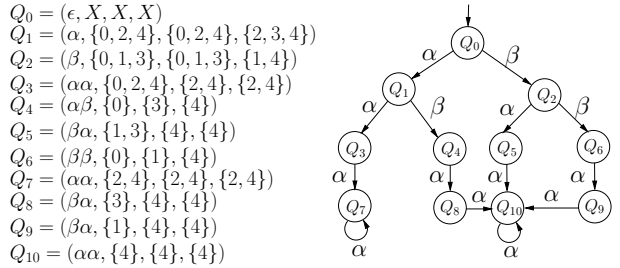

Fig. 4. Observation sequence-based 2-delay state estimator corresponding to DES G in Figure 1-d.

(iii) initial state $X_{K,0}^{observation} = (\epsilon, X_0, \ldots, X_0)$; and
(iv) state transition function $\delta_{K,obs}^{observation} : (\Sigma_{obs,\epsilon}^K \times 2^X \times \ldots \times 2^X) \times \Sigma_{obs} \rightarrow (\Sigma_{obs,\epsilon}^K \times 2^X \times \ldots \times 2^X)$ defined as follows: if the current state is the $(K+2)$-tuple $Q = (\Omega, Z_K, \ldots, Z_0) \in \Sigma_{obs,\epsilon}^K \times 2^X \times \ldots \times 2^X$, where $\Omega = \alpha_{-K} \ldots \alpha_{-1} \in \Sigma_{obs,\epsilon}^K$ and $Z_k \in 2^X, 0 \leq k \leq K$, then the next state for $\alpha_0 \in \Sigma_{obs}$ is $\hat{Q} = \delta_{K,obs}^{observation}(Q, \alpha_0) = (\hat{\Omega}, \hat{Z}_K, \ldots, \hat{Z}_0)$ where $\hat{\Omega} = \Omega\alpha_0/\alpha_{-K}$, and the sets $\hat{Z}_k$ are defined recursively as $\hat{Z}_k = \{z \mid z \in Z_{k-1}, \exists \hat{z} \in \hat{Z}_{k-1} : \hat{z} \in \delta_o(z, \alpha_{-k+1})\}$ for $k = 1, 2, \ldots, K$ with $\hat{Z}_0 = \delta_o(Z_0, \alpha_0)$.
[Note that AC denotes the states of this automaton that are accessible starting from initial state $X_{K,0}^{observation}$.] ■

Remark 14. The OS-KDE introduced in Definition 13 is related to the inverter with delay that was introduced in [16]. Assuming that the system is invertible with delay, the inverter in [16] acts as an *online* algorithm which, for a given time index, stores the K subsequent observations (where K is the fixed delay in the definition of invertibility with delay) in order to be able to refine the state estimate at this time index (using back propagation). The refined state estimate that is obtained is used along with the plant model to reconstruct the executed sequence of events. The observation sequence-based KDE is a finite structure that *captures* all estimates with delay for any observation sequence. In other words, what we do here can be seen as an *offline* approach for refining the current state estimate using any possible sequence of observations and K future observations. This is necessary when trying to verify system properties that depend on delayed state estimates (such as K-step opacity). ■

Example 15. Consider the DES in Figure 1-d. For this system, the observation sequence-based 2-delayed state estimator $G_{2,obs}^{observation}$ is represented in Figure 4. The initial state $X_0 = X$ and hence the OS-KDE initial-state $X_{2,0}^{observation}$ becomes $(\epsilon, X, X, X) \equiv Q_0$ which, using the notation in Definition 13, implies that $Z_0 = X$, $Z_1 = X$, $Z_2 = X$, $\Omega = \epsilon$ and $\alpha_{-1} = \alpha_{-2} = \epsilon$. Upon observing α ($\alpha_0 = \alpha$), the current (system) state estimate becomes $\hat{Z}_0 = \{2,3,4\}$ and since α can only be observed from $\{0,2,4\}$, $\hat{Z}_1 = \{0,2,4\}$. Also, since only one observation has been made, $\hat{Z}_2 = \hat{Z}_1 = \{0,2,4\}$; finally, $\hat{\omega} = \omega\alpha/\alpha_{-2} = \epsilon\alpha/\epsilon$, so that the next OS-KDE state upon observing α becomes $(\alpha, \{0,2,4\}, \{0,2,4\}, \{2,3,4\}) \equiv Q_1$.

Note that at OS-KDE state Q_1, $Z_0 = \{2,3,4\}$, $Z_1 = \{0,2,4\}$, $Z_2 = \{0,2,4\}$, $\Omega = \alpha$ and hence $\alpha_{-1} = \alpha$ and $\alpha_{-2} = \epsilon$. If β is observed at OS-KDE state Q_1, i.e., $\alpha_0 = \beta$, the next OS-KDE state $\hat{Q} = (\hat{\Omega}, \hat{Z}_2, \hat{Z}_1, \hat{Z}_0) \equiv Q_4$ can be obtained via

$$\begin{aligned}
\hat{Z}_0 &= \delta_o(Z_0, \alpha_0) = \delta_o(\{2,3,4\}, \beta) = \{4\} \\
\hat{Z}_1 &= \{z \in Z_0 \mid \exists \hat{z} \in \hat{Z}_0 \{\hat{z} \in \delta_o(z, \alpha_0)\}\} \\
&= \{z \in \{2,3,4\} \mid \exists \hat{z} \in \{4\} \{\hat{z} \in \delta_o(z, \beta)\}\} = \{3\} \\
\hat{Z}_2 &= \{z \in Z_1 \mid \exists \hat{z} \in \hat{Z}_1 \{\hat{z} \in \delta_o(z, \alpha_{-1})\}\} \\
&= \{z \in \{0,2,4\} \mid \exists \hat{z} \in \{3\} \{\hat{z} \in \delta_o(z, \alpha)\}\} = \{0\},
\end{aligned}$$

with $\hat{\Omega}$=$\Omega\alpha_0/\alpha_{-2}$=$\alpha\beta/\epsilon$=$\alpha\beta$.

OS-KDE state $Q_4 \equiv (\alpha\beta, \{0\}, \{3\}, \{4\})$ conveys the following information: the current state estimate is $\{4\}$, the previous state estimate is $\{3\}$, and the estimate of the system state two observations ago is $\{0\}$. Also $\alpha\beta$ captures the last two observations (observed in that order). Using this approach for the remaining possible observations, the observation sequence-based 2-delayed state estimator can be completed as shown in Figure 4 (states associated with empty state estimates of the form $(\Omega, \emptyset, \emptyset, \ldots, \emptyset)$ and transitions from/to these state have not been included). Note that the SM-KDE and OS-KDE in Figures 3-a and 3-b respectively are identical automata but, as clarified later on, this will not necessarily be the case in general. ■

In the sequel, we obtain a characterization of each set of states Z_k, $0 \leq k \leq K$, in the OS-KDE state $Q = (\Omega, Z_K, \ldots, Z_0)$ reached via a sequence of observations ω. Specifically, if $|\omega| \geq k$, we show that $j \in Z_k$, $0 \leq k \leq K$, if and only if there exists a string t in $L(G)$ that has projection ω, and visits state j exactly k observations ago; if there does not exist $t \in L(G)$ such that $P(t) = \omega$ then $Z_k = \emptyset$, $0 \leq k \leq K$. Furthermore, if $|\omega| < k$, then Z_k, $0 \leq k \leq |\omega|$, is as described above and $Z_k = Z_{|\omega|}$ for $|\omega| + 1 \leq k \leq K$. The following theorem states this formally; the proof is provided in [15].

Theorem 16. *Consider the OS-KDE constructed as in Definition 13 and suppose that state Q is reachable from the OS-KDE initial state $X_{K,0}^{observation} = (\epsilon, X_0, X_0, \ldots, X_0)$ via the string $\omega = \alpha_0\alpha_1 \ldots \alpha_n$. Then, the OS-KDE state Q can be characterized as follows:*
(i) $|\omega| < K$: $Q = (\Omega, Z_K, \ldots, Z_0) \in (\Sigma_{obs,\epsilon}^K, 2^X, \ldots, 2^X)$ with

1) $\Omega = \alpha_0 \dots \alpha_n$,
2) $Z_0 = \{j \in X \mid \exists t \in \Sigma^*, \exists i \in X_0 \{P(t) = \omega, j \in \delta(i,t)\}\}$,
3) *for* $1 \leq k \leq |\omega|$, $Z_k = \{j \in X \mid \exists t \in \Sigma^*, \exists t' \in \bar{t}, \exists i \in X_0 \{ P(t) = \omega,\ P(t)/P(t') = \alpha_{n-k+1} \dots \alpha_n,\ j \in \delta(i,t'),\ \delta(j, t/t') \neq \emptyset\}\}$, *and*
4) *for* $|\omega| + 1 \leq k \leq K$, $Z_k = Z_{|\omega|}$.

(ii) $|\omega| \geq K$: $Q = (\Omega, Z_K, \dots, Z_0) \in (\Sigma^K_{obs,\epsilon}, 2^X, \dots, 2^X)$ *with*

1) $\Omega = \alpha_{n-K+1} \dots \alpha_n$,
2) $Z_0 = \{j \in X \mid \exists t \in \Sigma^*, \exists i \in X_0 \{P(t) = \omega, j \in \delta(i,t)\}\}$,
3) *for* $1 \leq k \leq K$, $Z_k = \{j \in X \mid \exists t \in \Sigma^*, \exists t' \in \bar{t}, \exists i \in X_0 \{ P(t) = \omega,\ P(t)/P(t') = \alpha_{n-k+1} \dots \alpha_n,\ j \in \delta(i,t'),\ \delta(j, t/t') \neq \emptyset\}\}$.

(iii) $Z_k = \emptyset$, $0 \leq k \leq K$, *when there is no* $t \in L(G)$ *such that* $P(t) = \omega$. ■

The proof for the following corollary can also be found in [15].

Corollary 17. *The* k*-delayed state estimate* $\hat{X}_{|\omega|-k}(\omega)$, $0 \leq k \leq \min(K, |\omega|)$, *after observing* $\omega = \alpha_0 \alpha_1 \dots \alpha_n$ *is given by* $\hat{X}_{|\omega|-k}(\omega) = Z_k$ *where* $\delta^{observation}_{K,obs}(X^{observation}_{K,0}, \omega) = (\Omega, Z_K, \dots, Z_0)$. ■

Corollary 17 proves that the OS-KDE captures the set of all k-delayed state estimates, $0 \leq k \leq K$, via its $(K+2)$-tuples and hence, by Theorem 5, it can be used for verifying K-step opacity.

Corollary 18. *Discrete event system* G *is* (S, P, K)*-opaque if and only if for all* $Q = (\Omega, Z_K, \dots, Z_0) \in X^{observation}_{K,obs}$, $k \in \{0, \dots, K\}$,

$$Z_k \nsubseteq S \text{ or } Z_k = \emptyset, \tag{4}$$

where $X^{observation}_{K,obs}$ *is the set of states in* $G^{observation}_{K,obs}$ *that are reachable from the initial state* $X^{observation}_{K,0} = (\epsilon, X_0, \dots, X_0, X_0)$. ■

Example 19. As discussed in Example 12, DES G in Figure 1-d with $X_0 = \{0,1,2,3,4\}$ is not $(\{0\}, P, 2)$-opaque since observing the sequence of observations $\alpha\beta$ (or $\beta\beta$) reveals that the system was in state 0 two steps ago and state 0 is a secret state. Note that in the observation sequence-based 2-delay state estimator (depicted in Figure 4) the states reachable via $\alpha\beta$ (or $\beta\beta$) are $Q_4 = (\alpha\beta, \{0\}, \{3\}, \{4\})$ (or $Q_6 = (\beta\beta, \{0\}, \{1\}, \{4\})$). The 2-delayed state estimate associated with either of these states is $\{0\}$ and falls entirely within the set of secret states, which indicates that the system is not $(\{0\}, P, 2)$-opaque. ■

C. Analysis of State-Space Complexity for K-Delay State Estimators

1) State Complexity of OS-KDE: The construction of $G^{observation}_{K,obs}$ suggests that its number of states could be as high as $(|\Sigma_{obs}|+1)^K \times (2^N)^{(K+1)}$, where N denotes the number of states of the given automaton G. However, as we argue next, its state complexity is $O((|\Sigma_{obs}|+1)^K \times 2^N)$ which is significantly lower.

Theorem 20. *Given a non-deterministic finite automaton* $G = (X, \Sigma, \delta, X_0)$ *and a natural projection map* P *with respect to the set of observable events* Σ_{obs} *(*$\Sigma_{obs} \subseteq \Sigma$*), the state complexity of* $G^{observation}_{K,obs}$ *(constructed according to Definition 13) is* $O((|\Sigma_{obs}| + 1)^K \times 2^N)$, *where* $N = |X|$ *denotes the number of states of the given automaton* G. ■

Proof: We establish the state space complexity of $G^{observation}_{K,obs}$ by observing that given (i) the sequence of the past K observations $\Omega = \alpha_{-K+1} \dots \alpha_0$, and (ii) the K-delayed state estimate Z_K, the intermediate k-delayed state estimates Z_k $(0 \leq k < K)$ can be reconstructed uniquely using our knowledge of the plant model. First, note that given (i) and (ii), the current-state estimate, by definition, is readily available via $Z_0 = \delta_o(Z_K, \Omega)$. Next, we construct the intermediate k-delayed state estimates Z_k $(0 < k < K)$ in two steps:

(1) construct K sets of states X_k $(0 \leq k < K)$ as the set of states reachable in G, from states in Z_K via a string that produces the sequence of observations $\alpha_{-K+1} \dots \alpha_{-k}$. Following the notation in (3), we have $X_k = \delta_o(Z_K, \alpha_{-K+1} \dots \alpha_{-k})$;

(2) update all X_k with their post observations $\alpha_{-k+1} \dots \alpha_0$ to construct the k-delayed state estimate Z_k. To accomplish this, we can use an approach similar to the recursive state transition function introduced in Definition 13: by definition, Z_0 is the same as X_0. Next, we start from X_1 and remove all those estimates that are not consistent with observing α_0 (i.e., their transitions do not generate observation α_0 or do not result in a state in Z_0); this way, we obtain Z_1. Then, we consider X_2 and remove all states y in X_2 from which α_{-2} cannot occur (i.e., states y from which $\delta_o(y, \alpha_{-2}) = \emptyset$) or states y for which α_{-2} leads to a state outside Z_1 (i.e., states which have been removed in previous steps). We can repeat this procedure for $X_3, \dots, X_{K-1}$ and $\alpha_{-3}, \dots, \alpha_{-K+1}$. Therefore, using only Ω and Z_K, we can construct all intermediate k-delayed state estimates Z_k that were explicitly stored at each state of the K-delayed estimator in our earlier construction. Note that for the case when less than K observations are available, i.e., when $|\Omega| < K$, a similar approach can be taken to construct the intermediate k-delayed state estimates, $0 \leq k < |\Omega|$. ■

Example 21. For the DES G in Example 15 with the sequence-based 2-delayed state estimator in Figure 4, the state $(\alpha\alpha, \{0,2,4\}, \{2,4\}, \{2,4\})$ can simply be represented by $(\alpha\alpha, \{0,2,4\})$. We can easily obtain the missing unit-delayed and current state estimates as

described above. Note that based on the above notation, in state $(\alpha\alpha, \{0,2,4\}, \{2,4\}, \{2,4\})$, we have $Z_2 = \{0,2,4\}$ and $\alpha_{-2}\alpha_{-1} = \alpha\alpha$. Using this, we can obtain $X_1 = \delta_o(Z_2, \alpha_{-2}) = \delta_o(\{0,2,4\}, \alpha) = \{2,3,4\}$ and $X_0 = \delta_o(Z_2, \alpha_{-2}\alpha_{-1}) = \delta_o(\{0,2,4\}, \alpha\alpha) = \{2,4\}$. We can then set $Z_0 = X_0$ and reflect the post observation ($\alpha_{-1} = \alpha$) made after the previous estimation, which updates X_1 to $Z_1 = \{2,4\}$. ■

The above discussion not only demonstrates that storing the intermediate state estimates is not necessary (as long as the plant model is readily available and one is willing to do some computation) but also implies that keeping this information as part of the state label does not generate new states (even if the plant model is unavailable). In other words, $G^{observation}_{K,obs}$ with reduced state labels (constructed without explicitly storing the intermediate delayed state estimates as described above) is isomorphic to the one that stores them explicitly (described in Definition 13).

2) State Complexity of SM-KDE: Each state of the state mapping-based K-delay state estimator $G_{K,obs}$ is a $(K+1)$-dimensional state mapping. This suggests that the state complexity of this automaton is bounded by $2^{N^{K+1}}$ since there are N^{K+1} $(K+1)$-dimensional state mappings over the N states of the given automaton (each SM-KDE state is associated with a subset of these state mappings). In this section, we use the results on the state complexity of $G^{observation}_{K,obs}$ that we established in the beginning of this section to prove that the state complexity of $G_{K,obs}$ is $O((|\Sigma_{obs}|+1)^K \times 2^N)$. More specifically, we introduce a function which maps each state of $G^{observation}_{K,obs}$ to a state in $G_{K,obs}$ and then show that the range of this function covers all states of $G_{K,obs}$. This implies that the number of states of $G_{K,obs}$ is less than or equal to the number of states of $G^{observation}_{K,obs}$ and hence establishes that the state space complexity of $G_{K,obs}$ is also $O((|\Sigma_{obs}|+1)^K \times 2^N)$. The following theorem states and proves this formally. Due to space limitations, only a sketch of the proof is provided. The detailed proof can be found in [15].

Theorem 22. *Given a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$ and a natural projection map P with respect to the set of observable events Σ_{obs} ($\Sigma_{obs} \subseteq \Sigma$), the state complexity of $G_{K,obs}$ (constructed according to Definition 6) is $O((|\Sigma_{obs}|+1)^K \times 2^N)$, where N denotes the number of states of the given automaton G.* ■

Sketch of the Proof: To prove that the set of states of the SM-KDE, denoted by $X_{K,obs}$, has cardinality equal to or less than the set of states of the OS-KDE, denoted by $X^{observation}_{K,obs}$, we define a function $f : X^{observation}_{K,obs} \rightarrow 2^{X^{(K+1)}}$ and show that for all $Q \in X^{observation}_{K,obs}$:

(a) $f(Q) \in X_{K,obs}$ and,
(b) for each $m \in X_{K,obs}$, there exists at least one $Q \in X^{observation}_{K,obs}$ such that $f(Q) = m$.

The establishment of these two properties proves that the number of elements in the set $X_{K,obs}$ is less than or equal to the number of elements in the set $X^{observation}_{K,obs}$.

We define the mapping $f : X^{observation}_{K,obs} \rightarrow 2^{X^{(K+1)}}$ as follows:

i) For $\Omega = \alpha_0\alpha_1 \ldots \alpha_n$, $|\Omega| < K$, $Q = (\Omega, Z_{|\Omega|}, \ldots, Z_{|\Omega|}, Z_{|\Omega|}, Z_{|\Omega|-1}, \ldots, Z_0) \in X^{observation}_{K,obs}$:

$$f(Q) \equiv \{(j_0, j_1, \ldots, j_K) \mid (K - |\Omega| \leq k \leq K,$$
$$0 \leq l \leq |\Omega| - 1, 0 \leq w \leq K - |\Omega| - 1) : j_k \in Z_{K-k},$$
$$j_{w+1} = j_w, \exists t_l \in \Sigma^*\{P(t_l) = \alpha_l,$$
$$j_{K-|\Omega|+l+1} \in \delta(j_{K-|\Omega|+l}, t_l)\}\}.$$

ii) For $\Omega = \alpha_{n-K+1}\alpha_{n-K+2} \ldots \alpha_n, |\Omega| = K$, $Q = (\Omega, Z_K, \ldots, Z_0) \in X^{observation}_{K,obs}$:

$$f(Q) \equiv \{(j_0, j_1, \ldots, j_K) | (0 \leq k \leq K, 0 \leq l \leq K-1) :$$
$$j_k \in Z_{K-k}, \exists t_l \in \Sigma^*\{P(t_l) = \alpha_{n-K+1+l},$$
$$j_{l+1} \in \delta(j_l, t_l)\}\}.$$

Function f maps an OS-KDE state Q to a set of $(K+1)$-tuples of states $(j_0, \ldots, j_K)$ such that each state j_k in this tuple is chosen from the corresponding delayed state estimate Z_{K-k} and also such that the system can visit the sequence of states $j_0, \ldots, j_K$ and produce the sequence of observations Ω. We can show the function f satisfies property (a) by using the characterization of Q in Theorem 16 and property (b) by using the characterization of m in Theorem 9. ■

Note that for DES G in Figure 1-d, the number of states for both estimators is the same (indeed they are isomorphic) which shows that the introduced mapping between the states of these two estimators can be one-to-one. Note, however, that this is not necessarily the case. Consider, for example, the DES G in Figure 5-a with initial state set $X_0 = \{0,1\} = X$. Figures 5-b and 5-c depict $G^{observation}_{1,obs}$ and $G_{1,obs}$, respectively. As can be seen, storing the last observation results in creating more states compared to storing the 2-dimensional state mapping corresponding to the last 2 states visited.

Remark 23. The exponential complexity of the algorithms proposed in this paper for verifying K-step opacity is not desirable for implementation purposes. However, in [6], we showed that deciding whether the non-deterministic finite automaton G is K-step opaque is NP-hard for $|\Sigma_{obs}| > 1$. This implies that it is unlikely that any algorithm can verify K-step opacity in polynomial time. For more details, refer to [6]. ■

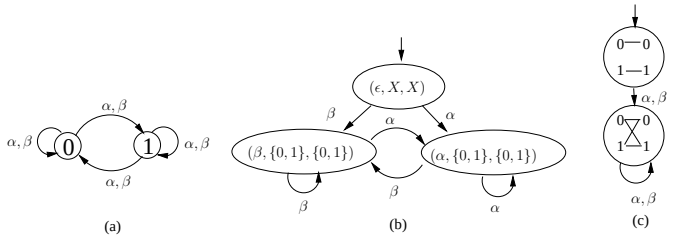

Fig. 5. Example demonstrating that the number of states in $G_{K,obs}$ can be less than the number of states in $G_{K,obs}^{observation}$: (a) G; (b) $G_{1,obs}^{observation}$; (c) $G_{1,obs}$.

V. Role of Delay in K-Step Opacity

In this section, we show that for $K' > K \geq 2^{N^2} - 1$, K-step opacity and K'-step opacity are equivalent. Note that K-step opacity does not in general imply K'-step opacity for $K' > K$ (in fact, Example 12 demonstrates this for the system in Figure 1-d) though the converse is trivially true (K'-step opacity implies K-step opacity for $K' > K$).

The idea behind the proof is the following: fix a point in the system's state trajectory. In the K-step opacity problem we are interested in finding how much we can say regarding the membership of the state, at that fixed point in time, to the set of secret states, after we make K additional observations. We can gain insight to this question by considering the estimate of the state at this fixed point as the *initial uncertainty* for an *initial-state estimation* problem. In [5], we studied the problem of initial-state estimation for a non-deterministic finite automaton: given a sequence of observations $\omega = \alpha_0\alpha_1\ldots\alpha_n$ and a set of possible initial states X_0, initial-state estimation requires the enumeration of all states that belong to X_0 and that could have generated this sequence of observations ω. We called these states the *initial state estimate* $\hat{X}_0(\omega)$ associated with the sequence of observations ω. In [5], we employed state mappings and showed that $\hat{X}_0(\omega) = m(1)$ where $m = m_0 \circ M(\alpha_0) \circ M(\alpha_1) \ldots \circ M(\alpha_n)$ and $m_0 = \odot_2(X_0)$. Our analysis in [5] resulted in an *initial-state estimator* (ISE), i.e., a deterministic finite automaton that is driven by observable events and whose states are essentially state mappings (its initial state is the state mapping $m_0 = \odot_2(X_0)$). More specifically, the set of starting states in the state mapping associated with the ISE state reached via string ω is the set of states from which a sequence of events that generates the observed sequence ω could have originated (i.e., $\hat{X}_0(\omega) = m(1)$ where m is the state mapping (state) reached from m_0 via ω). Since the ISE has at most 2^{N^2} states (because there are that many different state mappings for an automaton with $N = |X|$ states), we are guaranteed that each (reachable) ISE state can be reached via a string that generates at most K observations as long as $K \geq 2^{N^2} - 1$. Note that here we are concerned with the information conveyed by the *set* of all sequences of observations of length at most K, and not a specific sequence of observations.

Theorem 24. *Consider a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$ with $|X| = N$ and construct the K-delayed state estimators $G_{K,obs}$ and $G_{K^*,obs}$ for $K > K^* = 2^{N^2} - 1$ as described in Definition 6. Then, for any $(K+1)$-dimensional state mapping m associated with the SM-KDE state reached in $G_{K,obs}$ via $\omega = \alpha_0\alpha_1\ldots\alpha_n$ with $|\omega| > 2^{N^2} - 1$ and for each $m(k)$, $2^{N^2} \leq k \leq min(K, |\omega|)$, there exists a (K^*+1)-dimensional state mapping m' associated with a state reached in $G_{K^*,obs}$ via some $\omega' = \alpha_0\alpha_1\ldots\alpha_{n-k}\alpha'_{n-k+1}\ldots\alpha'_{n'}$ for some $n' \leq n + 2^{N^2} - 1 - k$ and with $\alpha'_{n-p} \in \Sigma_{obs}, k-1 \leq p \leq n-n'$, such that $m(k) = m'(k+n'-n)$.* ■

Proof: Recall that in any K-delay state estimator $G_{K,obs}$, the k-delayed state estimate due to observation ω is captured via the set $m(k)$, where m is the $(K+1)$-dimensional state mapping associated with the state reached in $G_{K,obs}$ via $\omega = \alpha_0\alpha_1\ldots\alpha_n$ (k satisfies $0 \leq k \leq min(K, |\omega|)$). Now consider the fixed point in time after the sequence of observations $\alpha_0\alpha_1\ldots\alpha_{n-k}$ has been observed. Once k more observations are made (i.e., once $\alpha_{n-k+1}\alpha_{n-k+2}\ldots\alpha_n$ are observed), the set $m(k)$ denotes the k-delayed state estimate at that fixed point due to the sequence of observations $\omega = \alpha_0\alpha_1\ldots\alpha_{n-k}\alpha_{n-k+1}\ldots\alpha_n$. Similarly, $m'(l)$ denotes the l-delayed state estimates of that fixed point due to the sequence of observations $\omega' = \alpha_0\alpha_1\ldots\alpha_{n-k}\alpha'_{n-k+1}\ldots\alpha'_{n'}$ for $n'-l = n-k$. In other words, $m(k)$ represents the k-delayed state estimate, if after the passage of the system through the state at that fixed point $\alpha_{n-k+1}\alpha_{n-k+2}\ldots\alpha_n$ is observed, whereas $m'(l)$ denotes the l-delayed state estimate at this same point if $\alpha'_{n-k+1}\ldots\alpha'_{n'}$ is observed. To prove Theorem 24, we need to show that assuming $k \geq 2^{N^2}$, there exists an $l \leq 2^{N^2} - 1$ such that the l-delayed state estimate at that same fixed time due to a shorter sequence of observations $\omega' = \alpha_0\alpha_1\ldots\alpha_{n-k}\alpha'_{n-k+1}\ldots\alpha'_{n'}$ with $n' = l+n-k$ is the same as the k-delayed state estimate of that fixed point due to the sequence of observations $\omega = \alpha_0\alpha_1\ldots\alpha_{n-k}\alpha_{n-k+1}\ldots\alpha_n$.

Denote the estimate of the system's (current) state at that point (i.e., the estimate after observing $\alpha_0\alpha_1\ldots\alpha_{n-k}$) by $Z \subseteq X$. The problem of k-delayed estimation of the state of the system at the fixed point in time after observing $\omega = \alpha_0\alpha_1\ldots\alpha_n$ can be viewed as an initial-state estimation problem where (due to the observations $\alpha_0\alpha_1\ldots\alpha_{n-k}$ that have been made before

reaching that fixed point) the initial uncertainty about the "initial state" is the set Z. Hence, the set $m(k)$ after observing $\omega = \alpha_0\alpha_1 \ldots \alpha_n$ is the same as the set of starting states of the state mapping that is associated with the state that is reached via $\alpha_{n-k+1}\alpha_{n-k+2} \ldots \alpha_n$ in the ISE whose initial state is associated with the state mapping $\odot_2(Z)$. Note that the string $\alpha_{n-k+1}\alpha_{n-k+2} \ldots \alpha_n$ has length $k > 2^{N^2} - 1$. Since the ISE has at most 2^{N^2} states, strings of length at most $2^{N^2} - 1$ can be chosen to visit any (reachable) ISE state. This implies that the state reached in this ISE via the string $\alpha_{n-k+1}\alpha_{n-k+2} \ldots \alpha_n$ of length k can also be reached via a string of length less than or equal to $2^{N^2} - 1$, which we denote by $\alpha'_{n-k+1}\alpha'_{n-k+2} \ldots \alpha'_{n'}$ for some $n' \leq n + 2^{N^2} - 1 - k$. Since the states reached in the ISE via either of these strings are identical, the k-delayed state estimate due to ω is the same as the $(k-(n-n'))$-delayed state estimate due to ω'. This completes the proof. ■

The above result can be used to show that K'-step opacity is equivalent to K-step opacity for $K' > K \geq 2^{N^2} - 1$. We prove this by showing that for $K \geq 2^{N^2}$, K-step opacity is equivalent to K^*-step opacity with $K^* = 2^{N^2} - 1$. The proof can be found in [15].

Theorem 25. *For a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$, K-step opacity is equivalent to K^*-step opacity for $K > K^* = 2^{N^2} - 1$ where $N = |X|$.* ■

VI. Conclusion

In this paper, we defined, analyzed, and characterized the notion of K-step opacity for discrete event systems that can be modeled as non-deterministic finite automata. The notion of K-step opacity, for $K \geq 0$, requires that the entrance of the system to a set of secret states S, at any time during the past K observations, remains opaque (uncertain) to outsiders. To verify K-step opacity, we introduced the K-delay state estimator which provides K-delayed state estimates. These are the estimates of the state of the system k observations ago ($0 \leq k \leq K$) and are consistent with all observations so far (including the last k observations). We show that for a system to be K-step opaque, all k-delayed state estimates (associated with states of the K-delay state estimator) need to contain at least one state outside the secret set S. The proposed verification method has state complexity $O((|\Sigma_{obs}|+1)^K \times 2^N)$, where N denotes the number of states of the given automaton G.

References

[1] R. Focardi and R. Gorrieri, "A taxonomy of trace–based security properties for CCS," in *Proc. of the 7th Workshop on Computer Security Foundations*, June 1994, pp. 126–136.

[2] S. Schneider and A. Sidiropoulos, "CSP and anonymity," in *Proc. of the 4th European Symposium on Research in Computer Security*, September 1996, pp. 198–218.

[3] J. Bryans, M. Koutny, L. Mazare, and P. Ryan, "Opacity generalised to transition systems," *International Journal of Information Security*, vol. 7, no. 6, pp. 421–435, November 2008.

[4] A. Saboori and C. N. Hadjicostis, "Notions of security and opacity in discrete event systems," in *Proc. of the 46th IEEE Conference on Decision and Control*, December 2007, pp. 5056–5061.

[5] ——, "Verification of initial–state opacity in security applications of DES," in *Proc. of the 9th International Workshop on Discrete Event Systems*, May 2008, pp. 328–333.

[6] ——, "Verification of K-step opacity and analysis of its complexity," in *Proc. of the 48th IEEE Conference on Decision and Control*, December 2009, pp. 205–210.

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

[8] E. Badouel, M. Bednarczyk, A. Borzyszkowski, B. Caillaud, and P. Darondeau, "Concurrent secrets," *Discrete Event Dynamic Systems*, vol. 17, no. 4, pp. 425–446, December 2007.

[9] J. Dubreil, P. Darondeau, and H. Marchand, "Opacity enforcing control synthesis," in *Proc. of the 9th International Workshop on Discrete Event Systems*, May 2008, pp. 28–35.

[10] N. Hadj-Alouane, S. Lafrance, L. Feng, J. Mullins, and M. Yeddes, "On the verification of intransitive noninterference in multilevel security," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 5, pp. 948–958, October 2005.

[11] C. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Kluwer Academic Publishers, 2008.

[12] J. S. Meditch, "A survey of data smoothing for linear and nonlinear dynamic systems," *Automatica*, vol. 9, no. 2, pp. 151–162, March 1973.

[13] A. Saboori and C. N. Hadjicostis, "Verification of infinite-step opacity and analysis of its complexity," in *Proc. of the 2009 Workshop on Dependable Control of Discrete Systems*, June 2009, pp. 51–56.

[14] TAKOS: A Java toolbox for analyzing the K-opacity of systems. [Online]. Available: http://toolboxopacity.gforge.inria.fr/

[15] A. Saboori and C. N. Hadjicostis, "Supplement material for the paper "Verification of K-step opacity and analysis of its complexity"," to be made available: http://www.eng.ucy.ac.cy/hadjicostis/publications.html.

[16] C. M. Özveren and A. S. Willsky, "Invertibility of discrete event dynamic systems," *Mathematics of Control, Signals, and Systems*, vol. 5, no. 4, pp. 365–390, July 1992.