2016 American Control Conference (ACC)
Boston Marriott Copley Place
July 6-8, 2016. Boston, MA, USA

A Framework for Opacity in Linear Systems*

Bhaskar Ramasubramanian[1], Rance Cleaveland,[2] and Steven I. Marcus[1]

***Abstract*— We present a framework for opacity in cyberphysical systems modeled as discrete time linear time invariant systems. A set of secret states is k*−ISO* with respect to a set of nonsecret states if, starting from these sets at time 0, the outputs at time k are indistinguishable to a passive adversarial observer. Necessary and sufficient conditions for k−ISO are given in terms of reachable sets of the system. Properties of k−ISO under unions and intersections are verified. It is seen that while unions of opaque sets preserve opacity, this is not necessarily true for intersections. We show that under certain conditions, k−ISO is equivalent to output controllability. Finally, we present an algorithm to compute a k−ISO set of states, given candidate secret and nonsecret sets of initial states.**

I. Introduction

Cyberphysical systems (CPS) are entities in which the working of the physical system is intimately linked to the functioning of computers that influence the interactions between the system and a controller, or among subsystems. Since these systems are often controlled via a network, computational resources and bandwidth also affect their working. Examples of large scale CPS include power systems, water distribution networks, and medical devices. While computer controlled systems are more efficient, the sharing of information among devices and across geographies makes the system vulnerable to attacks. An attack could be carried out on the physical system itself, on the computer controlling the system, or on the communication links between the system and the computer. A compilation of exposed vulnerabilities in some existing systems, and means of mitigating threats can be found in [1], [2], [3]. While the aforementioned places emphasis on the attacker's abilities, one could also choose to focus on the flow of information from the CPS to the attacker [4], [5].

Opacity is an instance of the latter, and is a property that captures whether an intruder, modeled as a passive observer, can infer a "secret" of a system based on its observation of the system behavior. The current state of the art in this area studies opacity within the framework of discrete event systems (DES) described by regular languages [6], [7]. Techniques from supervisory control can be used to enforce opacity on a system [8], [9]. In other words, a controller can be designed to disable actions that lead to the leaking of the secret.

Although this theory is quite rich, it suffers from the drawback that the states in a DES are discrete. In many practical systems, it is common for the states to take values in a continuous domain. This is indeed the case in CPS such as power systems and water networks. This paper considers CPS modeled as a discrete time linear time invariant (DT-LTI) system [10] (thus, while time steps are discrete, the state, control, and output variables are real valued). We use tools from control theory to study opacity for such systems.

We define a notion of opacity for DT-LTI systems and establish conditions to achieve opacity in terms of sets of reachable states. Opacity of a given DT-LTI system is shown to be equivalent to the output controllability of a system obeying

*This work was supported by the National Science Foundation under Grant CNS−1446665.

[1] Department of Electrical and Computer Engineering and Institute for Systems Research, University of Maryland, College Park, MD 20742, USA.
{rbhaskar, marcus}@umd.edu

[2] Department of Computer Science and Institute for Systems Research, University of Maryland, College Park MD 20742, USA. rance@cs.umd.edu

978-1-4673-8682-1/$31.00 ©2016 AACC

the same dynamics, but with different initial conditions. Finally, we present an algorithm to determine an opaque set of states, given candidate sets of secret and nonsecret states.

A. Related Work

Opacity was first presented as a tool to study cryptographic protocols in [11]. The intruder was a passive observer who could read messages exchanged between two parties, but could not modify, block, or send a message. The aim of the parties was to exchange secret information without making it obvious to the intruder. A theory of supervisory control for DES represented by finite state automata (FSA) and regular languages was formulated in [12], [13]. This framework spawned research in many areas including fault diagnosis [14], hybrid systems [15], and robotics [16].

DES were used to study opacity in [6], which assumed multiple intruders with different observation maps. Assuming the supervisor could control all events, it was shown that there exists an optimal control that enforced opacity. The secret can be specified by a subset of states or sublanguages of the DES. Opacity can be defined for each instance accordingly. Verification of the opacity of a secret specified as a language was presented in [8], [17], while [7], [18], [19] studied the same for secrets specified as states. Opacity was compared with detectability and diagnosability of DES and other privacy properties like secrecy and anonymity in [20]. The enforcement of opacity using supervisory control was studied in [8], [9]. [21] formulated an alternate method of opacity enforcement using insertion functions, which are entities that modify the output behavior of the system in order to maintain the secret. The authors also proposed a notion of joint opacity, where a system can be observed by observers who share their observations with a coordinator, which then verifies opacity.

B. Outline of Paper

Section (II) gives a brief introduction to opacity for discrete event systems. We formulate and motivate our definition of opacity for LTI systems in Section (III). Section (IV) gives examples illustrating our framework. In Section (V), we establish necessary and sufficient conditions to establish $k-$ISO in terms of reachable sets, while Section (VII) shows that $k-$ISO is equivalent to output controllability of a slightly modified LTI system. Section (VI) verifies $k-$ISO under unions and intersections of sets. We propose an algorithm to compute an opaque set, given candidate secret and nonsecret sets of states in Section (VIII). We conclude the paper with ideas on future areas of research in this topic in Section (IX).

II. Opacity for Discrete Event Systems

In this section, we review opacity for discrete event systems. The reader is referred to [7], [20], [22] for a detailed exposition.

Let $G=(X,\Sigma,f,X_0)$ be an FSA, where X is a nonempty set of states, $X_0 \subseteq X$ is a nonempty set of initial states, and Σ is the set of events. $f: X \times \Sigma \rightarrow X$ is the (partial) state transition function: given $x,y \in X$ and $\sigma \in \Sigma$, we write $f(x,\sigma)!$ if $f(x,\sigma)=y$ is a valid transition. The transition function is extended to $f: X \times \Sigma^* \rightarrow X$ in the usual way. The language generated by G is $\mathscr{L}(G) := \{s \in \Sigma^* : f(x,s)!\}$, and describes all possible trajectories of the system. Let K_1 and K_2 be sublanguages of $\mathscr{L}(G)$. Let $P: \Sigma^* \rightarrow \Sigma^*$ be a projection map. Then, if $s \in \Sigma^*$ occurs in the system, an external agent would see $P(s)$.

Definition 2.1: K_1 is *strongly language based opaque (LBO)* with respect to K_2 and P if for every trajectory in K_1, there exists a trajectory in K_2 that 'looks' the same under P, i.e. $K_1 \subseteq P^{-1}(P(K_2))$.

Definition 2.2: K_1 is *weakly LBO* with respect to K_2 and P if there exists a trajectory in K_1 that is confused with some trajectory in K_2, under P, i.e. $K_1 \cap P^{-1}(P(K_2)) \neq \phi$.

Definition 2.3: Given G with $X_s, X_{ns} \subseteq X_0$, and P, X_s is *initial state opaque (ISO)* with respect to X_{ns} and P if for every $i \in X_s$ and every $t \in L(G,i)$ such that $f(i,t)!$, there exists $j \in X_{ns}$ and $t' \in L(G,j)$ such that $f(j,t')!$, and $P(t)=P(t')$.

These definitions are essentially equivalent, as there exist polynomial time algorithms that relate any pair of the notions of opacity [22].

III. Opacity for LTI Systems

Consider the DT-LTI system:

$$\begin{aligned} x(t+1) &= Ax(t)+Bu(t) \\ x(0) &= x_0 \in X_0 \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n, u \in \mathbb{R}^m, y \in \mathbb{R}^p$, and A,B,C are real matrices of appropriate dimensions. The subscripts s and ns, when appended to the variables will respectively correspond to trajectories starting from secret and nonsecret initial states. Let $\mathscr{K}$ be a set of positive integers corresponding to the time instants the adversary will observe the system.

Definition 3.1: For (1), given $X_s, X_{ns} \subseteq X_0$ and $\mathscr{K}$, X_s is *strongly k−ISO* with respect to X_{ns} if for all $k \in \mathscr{K}$, for all $x_s(0) \in X_s$ and for every sequence of admissible controls $u_s(0),\ldots,u_s(k)$, there exist an $x_{ns}(0) \in X_{ns}$, and a sequence of admissible controls $u_{ns}(0),\ldots,u_{ns}(k)$ such that $y_s(k) = y_{ns}(k)$.
This means that, starting from any secret state, and applying any sequence of k admissible controls corresponding to the instants the adversary makes an observation, the system will reach a state that is indistinguishable from a state reached by the application of some admissible control sequence of length k, starting from some nonsecret state. A weaker notion of k−ISO can be similarly defined.

Definition 3.2: X_s is *weakly k−ISO* w.r.t. X_{ns} if for all $k \in \mathscr{K}$, there exist $x_s(0) \in X_s$, a sequence of admissible controls $u_s(0),\ldots,u_s(k)$, $x_{ns}(0) \in X_{ns}$, and a sequence of admissible controls $u_{ns}(0),\ldots,u_{ns}(k)$ such that $y_s(k) = y_{ns}(k)$.

Motivation for Definition

This notion of opacity for LTI systems is different from familiar definitions of observability. The observability problem aims to determine the initial state $x(0)$, given the entire output and control history. Here, however, the adversary has access to only snapshots of the system output, and it must determine $x(0)$ from these. The small number of observations of the system is motivated by the fact that an adversary might not want to reveal its presence to the system (the 'passive' nature of the intruder only prevents it from performing actions detrimental to the system; the system can take corrective action to ensure opacity if it detects the adversary). In this paper, we shall assume that the adversary makes exactly one observation, and at time k, i.e. $\mathscr{K} = \{k\}$. The case when the adversary makes observations at different times, and limiting the number of observations it will be allowed to make is a subject for future work.

Our formulation differs from definitions of opacity in the DES literature where the observation of the entire secret trajectory must coincide with the observation of a non-secret trajectory. Here, we only need that the secret and nonsecret outputs at time k coincide. This is again motivated by the fact that the adversary might not want to reveal itself to the system by making multiple observations. k−ISO also differs from *k−step opacity* proposed in [23]. In their formulation, k−step opacity is achieved when the adversary does not know if the system entered a secret state in the k previous steps. We require that the ambiguity exists only at time k. We demonstrate that an additional requirement to our conditions for k−ISO will also guarantee k−step opacity.

The adversary is assumed to have knowledge of the initial sets of secret and nonsecret states. Recall that the aim here is that, following an observation at time k, the adversary should not be able to infer whether the system started from a secret state or a nonsecret state at time 0. From the system's point of view, it needs to know how much information it can reveal about itself without allowing the adversary to meet its goal. From the adversary's perspective, it needs to know whether it has deduced the secret at time k, which would entail some knowledge of the set of 'solutions' to the problem. This assumption strikes a balance between these two requirements. Variants of this problem with other assumptions on the information available to the adversary will be studied in the future.

Controls form an integral part of the definition of $k-$ISO, which automatically allows for simultaneously verifying and determining policies to enforce opacity. This differs from the DES framework, where opacity enforcing supervisory control is treated separately from the verification of opacity.

Our definition of $k-$ISO for LTI systems is also different from the notion of simulation relations between dynamical systems [24]. In simulation relations, we typically verify the 'equality' of two systems governed by different dynamics. In our framework however, we try to identify equivalence classes of outputs at time k, and opacity is deemed to have been achieved if the system starting from two disjoint sets of states at time 0 reaches the same 'equivalence class' of outputs at time k.

IV. Examples

Example 4.1: Let $X_s, X_{ns} \subseteq X_0$ with $X_s = \{\begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T, \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T\}$ and $X_{ns} = \mathbb{R}^3 \setminus X_s$. Let $A = I_{3\times 3}$, $B = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}^T$ and $C = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$. The output for the dynamics in (1) for $x_s(0) = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T$ is:

$$y_s(i) = 1 + 3\sum_{j=0}^{i-1} u_s(j) \tag{2}$$

$x_s(0) = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$ will also give the same $y_s(i)$. Now, let $x_{ns}(0) = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^T$. From (1), we have

$$y_{ns}(i) = 1 + 3\sum_{j=0}^{i-1} u_{ns}(j) \tag{3}$$

Comparing (2) and (3), X_s will be strongly $k-$ISO w.r.t. X_{ns} if for every admissible control sequence $\{u_s(0), \ldots, u_s(k-1)\}$, there is an admissible control sequence $\{u_{ns}(0), \ldots, u_{ns}(k-1)\}$ such that: $\sum_{j=0}^{k-1} u_s(j) = \sum_{j=0}^{k-1} u_{ns}(j)$.

Example 4.2: Secure movement of money from a bank to an ATM is an interesting example. One way of ensuring security, from the bank's perspective, is to use heavily armored trucks. However, such customizations can be very expensive, and need to be continuously updated to stay ahead of potential attackers. Another method is to employ a set of identical dummy trucks. Assuming that the cost of carrying out an attack is reasonably high, attacking a truck carrying money is akin to playing the lottery. The goal (of the bank) is to ensure that an adversary cannot determine whether an observed truck is carrying money.

Let the set of secret states be the locations from which trucks carrying money originate. The non-secret states can be the whole space, excluding this set, or a predefined set of states.

Let the states of the system be the position and velocity of a truck and the control input be the acceleration. Assuming unit mass and unit sampling interval, a simplified DT-LTI model of the system is:

$$\begin{pmatrix} p(k+1) \\ v(k+1) \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} p(k) \\ v(k) \end{pmatrix} + \begin{pmatrix} 0.5 \\ 1 \end{pmatrix} a(k)$$
$$y(k) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} p(k) \\ v(k) \end{pmatrix} \tag{4}$$

The position at time k is given by:

$$p(k) = p(0) + kv(0) + \sum_{j=0}^{k-1} (k - j - 0.5) a(j) \tag{5}$$

Let this be the observation of the adversary at time k. The acceleration of a truck is typically upper bounded, i.e. $a(k) \leq a_{\max}$. Now, given $X_s(0), X_{ns}(0), a_{\max}$, and $p(k)$, the truck's initial position will be opaque to the adversary if for every control sequence starting from every $(p_s(0), v_s(0)) \in X_s(0)$, there exists a control sequence starting from some $(p_{ns}(0), v_{ns}(0)) \in X_{ns}(0)$ such that the positions of the trucks at time k are the same.

V. Opacity Using Reachable Sets

Let $U_s^k := \{u_{s0}, \ldots, u_{s(k-1)}\}$ and $U_{ns}^k := \{u_{ns0}, \ldots, u_{ns(k-1)}\}$. Let $X_s(k)$ and $X_{ns}(k)$ denote the sets of reachable states in k steps, starting from nonempty sets X_s and X_{ns} respectively.

$$X_s(k) = \bigcup_{x_0 \in X_s} \bigcup_{U_s^k} \{x : x(i+1) = Ax(i) + Bu(i), \forall i < k\}$$
$$X_{ns}(k) = \bigcup_{x_0 \in X_{ns}} \bigcup_{U_{ns}^k} \{x : x(i+1) = Ax(i) + Bu(i), \forall i < k\}$$

Theorem 5.1: X_s is strongly $k-$ISO with respect to X_{ns} if and only if $CX_s(k) \subseteq CX_{ns}(k)$.

Proof: First, let strong $k-$ISO hold. Then, for all $x_s(0) \in X_s$ and all $\{u_s(\cdot)\}$, there exist $x_{ns}(0) \in X_{ns}$ and $\{u_{ns}(\cdot)\}$ such that $y_s(k) = y_{ns}(k)$. Now, starting from X_s (respectively, X_{ns}) and applying k admissible controls, one reaches a state in $X_s(k)$ ($X_{ns}(k)$). Therefore, $k-$ISO ensures that for every $x_s(k) \in X_s(K)$, there exists $x_{ns}(k) \in X_{ns}(k)$ such that $y_s(k) = y_{ns}(k)$. This gives $CX_s(k) \subseteq CX_{ns}(k)$.

Now, let $CX_s(k) \subseteq CX_{ns}(k)$. Then, for every $x_s(k) \in X_s(k)$, there exists $x_{ns}(k) \in X_{ns}(k)$ such that $y_s(k) = y_{ns}(k)$. Since $X_s(k)$ and $X_{ns}(k)$ are reachable sets starting from X_s and X_{ns} respectively, this is equivalent to: for every $x_s(0) \in X_s$ and every $\{u(\cdot)\}$, there exists $x_{ns}(0) \in X_{ns}$ and $\{u_{ns}(\cdot)\}$ such that $y_s(k) = y_{ns}(k)$. This, by definition, is strong $k-$ISO. ■

Remark 5.2: This result can be extended to verify $k-$step opacity proposed in [23] by postulating that $CX_s(i) \subseteq CX_{ns}(i)$ holds for $i = m, m-1, \ldots, m-k+1$ for $\mathcal{K} = \{m\}$.

Proposition 5.3: If $X_s(k) \subseteq X_{ns}(k)$, then X_s is strongly $k-$ISO with respect to X_{ns}.

Unlike Theorem (5.1), Proposition (5.3) only gives a sufficient condition for $k-$ISO. To see that this condition is not necessary, let $C = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$, and $X_s(k) = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T$ and $X_{ns}(k) = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^T$. Then, $CX_s(k) = CX_{ns}(k)$, establishing $k-$ISO, even though $X_s(k) \not\subseteq X_{ns}(k)$.

Similar results hold for weak $k-$ISO.

Theorem 5.4: X_s is weakly $k-$ISO with respect to X_{ns} if and only if $CX_s(k) \cap CX_{ns}(k) \neq \phi$.

Proposition 5.5: If $X_s(k) \cap X_{ns}(k) \neq \phi$, then X_s is weakly $k-$ISO with respect to X_{ns}.

VI. $k-$ISO Under Set Operations

Properties of $k-$ISO are studied under unions and intersections. The properties verified will be for strong $k-$ISO, unless otherwise mentioned. Proofs of some results are omitted for brevity. Let X denote the set of initial states, and $X(k)$ be the set of states reachable in k steps, starting from X.

Proposition 6.1: Given sets of initial states $X_1, X_2, \cdots \subseteq X$, the reachable set in k steps of their union is equal to the union of the reachable sets in k steps of each set of initial states. That is, $(\bigcup_i X_i)(k) = \bigcup_i X_i(k)$.

Proof: $x \in (\bigcup_i X_i)(k)$

$$\Leftrightarrow \exists x_0 \in (\bigcup_i X_i), \exists \{u(\cdot)\}, \text{ (1) holds } \forall i < k, x(k) = x$$
$$\Leftrightarrow [(\exists x_0 \in X_1 \wedge \exists \{u(\cdot)\}) \text{ s.t. } (x \in X_1(k))] \vee$$
$$[(\exists x_0 \in X_2 \wedge \exists \{u(\cdot)\}) \text{ s.t. } (x \in X_2(k))] \vee \ldots$$
$$\Leftrightarrow x \in \bigcup_i X_i(k)$$

■

Corollary 6.2: Given $X_1, X_2, \cdots \subseteq X$ and $C : \mathbb{R}^n \to \mathbb{R}^m$, $C(\bigcup_i X_i)(k) = \bigcup_i CX_i(k)$.

Proposition 6.3: If X_{s_i} is $k-$ISO with respect to X_{ns} for each i, then $\bigcup_i X_{s_i}$ is $k-$ISO with respect to X_{ns}.

Proof: X_{s_i} $k-$ISO w.r.t. $X_{ns} \forall i$

$$\Leftrightarrow CX_{s_i}(k) \subseteq CX_{ns}(k) \forall i$$
$$\Leftrightarrow \bigcup_i CX_{s_i}(k) \subseteq CX_{ns}(k)$$
$$\Leftrightarrow C(\bigcup_i X_{s_i}(k)) \subseteq CX_{ns}(k)$$
$$\Leftrightarrow \bigcup_i X_{s_i} \text{ is } k-\text{ISO w.r.t. } X_{ns}$$

■

Proposition 6.4: If X_s is $k-$ISO w.r.t. X_{ns_i} for each i, then X_s is $k-$ISO w.r.t. $\bigcup_i X_{ns_i}$

Proposition 6.5: Given sets of initial states $X_1, X_2, \cdots \subseteq X$, $(\bigcap_i X_i)(k) \subseteq \bigcap_i X_i(k)$.

Proof: $x \in (\bigcap_i X_i)(k)$

$$\Rightarrow \exists x_0 \in (\bigcap_i X_i), \exists \{u(\cdot)\}, \text{ (1) holds } \forall i < k, x(k) = x$$
$$\Rightarrow [(\exists x_0 \in X_1 \wedge \exists \{u(\cdot)\}) \text{ s.t. } (x \in X_1(k))] \wedge$$
$$[(\exists x_0 \in X_2 \wedge \exists \{u(\cdot)\}) \text{ s.t. } (x \in X_2(k))] \wedge \ldots$$
$$\Leftrightarrow x \in \bigcap_i X_i(k)$$

■

Corollary 6.6: Given $X_1, X_2, \cdots \subseteq X$ and $C : \mathbb{R}^n \to \mathbb{R}^m$, $C(\bigcap_i X_i)(k) \subseteq \bigcap_i CX_i(k)$.

Remark 6.7: The reverse inclusions need not hold in (6.5) and (6.6). Let $C = I$, $X_1 = X_s$ and $X_2 = X_{ns}$. $X_1 \cap X_2 = \emptyset$, but $X_1(k) \cap X_2(k)$ need not be empty. [1]

Proposition 6.8: If X_{s_i} is $k-$ISO with respect to X_{ns} for each i, then $\bigcap_i X_{s_i}$ is $k-$ISO with respect to X_{ns}.

Proposition 6.9: If X_s is $k-$ISO with respect to X_{ns_i} for each i, then $CX_s(k) \subseteq \bigcap_i CX_{ns_i}(k)$. However, in general, X_s is not $k-$ISO with respect to $\bigcap_i X_{ns_i}$.

Proof: X_s $k-$ISO w.r.t. $X_{ns_i} \forall i$

$$\begin{aligned} &\Leftrightarrow CX_s(k) \subseteq CX_{ns_i}(k) \forall i \\ &\Rightarrow CX_s(k) \subseteq \bigcap_i CX_{ns_i}(k) \end{aligned}$$

However, we can have $\bigcap_i X_{ns_i} = \emptyset$, which means $C(\bigcap_i X_{ns_i})(k)$ is undefined. ■

Proposition 6.10: If X_{s_i} is weakly $k-$ISO with respect to X_{ns} for each i, then $\bigcup_i X_{s_i}$ is weakly $k-$ISO with respect to X_{ns}.

Remark 6.11: If X_{s_i} is weakly $k-$ISO with respect to X_{ns} for each i, then $\bigcap_i X_{s_i}$ need not be weakly $k-$ISO with respect to X_{ns}. That is, given $CX_{s_i}(k) \bigcap CX_{ns_i}(k) \neq \emptyset \forall i$, if $\bigcap_i X_{s_i} = \emptyset$, then $C(\bigcap_i X_{s_i})(k) \bigcap CX_{ns}(k)$ will not be defined.

VII. $k-$ISO AND OUTPUT CONTROLLABILITY

The output of (1) at time k is given by: $y(k) = CA^k x(0) + \sum_{j=0}^{k-1} CA^{k-j-1} Bu(j)$.

Definition 7.1: A state x of (1) is *output controllable* on $[0, k_f]$ if there exists a control sequence $\{u(\cdot)\}$ that transfers the system from $x(0) = x$ to $y(k_f) = 0$.

Theorem 7.2: Let X_s be (strongly or weakly) $k-$ISO with respect to X_{ns}. Then there exists a state of (1) that is output controllable on $[0, k]$. Further, if $k-$ISO is established for the pair $(x_s(0), x_{ns}(0)) \in X_s \times X_{ns}$ (and appropriate control sequences $\{u_s(\cdot)\}$ and $\{u_{ns}(\cdot)\}$), then the control sequence $u(i) = u_s(i) - u_{ns}(i)$, $i = 0, 1, \ldots, k-1$, will achieve output controllability for the initial state $x(0) = x_s(0) - x_{ns}(0)$.

Proof: $k-$ISO implies $y_s(k) = y_{ns}(k)$ for appropriate $x_s(0), \{u_s(\cdot)\}, x_{ns}(0)$ and $\{u_{ns}(\cdot)\}$. Setting $x(0) = x_s(0) - x_{ns}(0)$ and $u(i) = u_s(i) - u_{ns}(i)$, $i = 0, 1, \ldots, k-1$ in the dynamics of (1) ensures $y(k) = 0$, thus achieving output controllability of the state $x(0) = x_s(0) - x_{ns}(0)$. ■

Theorem 7.3: Let (1) be output controllable in k steps for a set of states $X_{oc}(0) \setminus \{0\}$ and controls $\{U(\cdot)\}$. Let X_1 and X_2 be sets such that every $x_1 \in X_1$ can be written as $x + x_2$, where $x \in X_{oc}(0) \setminus \{0\}$ and $x_2 \in X_2$. Then, X_1 is strongly $k-$ISO with respect to X_2.

Proof: Output controllability ensures that:

$$y(k) = CA^k x(0) + \sum_{j=0}^{k-1} CA^{k-j-1} BU(j) = 0 \qquad (6)$$

For any control sequence $\{u_1(\cdot)\}$, the output at time k, starting from any $x_1(0) \in X_1$ is: $y_1(k) = CA^k x_1(0) + \sum_{j=0}^{k-1} CA^{k-j-1} Bu_1(j)$. The output at time k starting from $x_2(0) \in X_2$ with the control sequence $\{u_1(\cdot) - U(\cdot)\}$ is: $y_2(k) = CA^k x_2(0) + \sum_{j=0}^{k-1} CA^{k-j-1} B[u_1(j) - U(j)]$. Using the assumption that every $x_1 \in X_1$ can be written as $x + x_2$, where $x \in X_{oc}(0) \setminus \{0\}$, $x_2 \in X_2$, and equation (6), we get $y_1(k) = y_2(k)$.

Thus, for any $x_1 \in X_1$ and any control sequence starting from x_1, there exist $x_2 \in X_2$ and another control sequence such that the outputs after k steps are the same. This is strong $k-$ISO with $X_s = X_1$ and $X_{ns} = X_2$. ■

VIII. COMPUTING AN OPAQUE SUBSET

Algorithm (1) returns a subset of the set of initial states that is $k-$ISO with respect to its complement. (The output is actually a union of opaque subsets. However, $k-$ISO is preserved under unions). Algorithm (2) extends it to the case with candidate sets of initial secret and nonsecret states.

Lemma 8.1: Algorithm (1) is correct.

Proof: $X_s = \bigcup_i X_i$, $C(\bigcup_i X_i)(k) = \bigcup_i CX_i(k)$ (from (6.2)). $CX_s(k) \subseteq CX_{ns}(k)$, which is necessary and sufficient for $k-$ISO (from (5.1)). ■

[1] Recall that the definition of the reachable set in k steps assumes a nonempty initial set of states.

Algorithm 1 Determine a $k-$ISO subset of a given set of initial states

```
Input: Set of initial states $X$, positive integer $k$, system model (1)
Output: $X_s \subseteq X$, that is $k-$ISO w.r.t. $X \setminus X_s$
$X(k) := \bigcup_{x_0 \in X} \bigcup_{U^k} \{x : x(i+1) = Ax(i) + Bu(i), \forall i < k\}$ // reachable set in k steps
$CX(k) := \{y : y = Cx(k), x(k) \in X(k)\}$
$X_s = \{\bigcup_{i=1}^{r} X_i : X_i \subseteq X \forall i \in \{1,\dots,r\}, \bigcup_{i=1}^{r} CX_i(k) \subseteq C(X \setminus \bigcup_{i=1}^{r} X_i)(k)\}$;
$X_{ns} = X \setminus X_s$
```

Algorithm 2 Determine a $k-$ISO subset, given candidate secret and nonsecret sets

```
Input: (Disjoint) Sets of prospective secret and nonsecret states $X_s^p(0), X_{ns}^p(0)$, positive integer $k$, system model (1)
Output: $X_s^* \subseteq X_s^p(0)$ that is $k-$ISO w.r.t. $X_{ns}^*$
$X_s^p(k) = \bigcup_{x_0 \in X_s^p(0)} \bigcup_{U^k} \{x : x(i+1) = Ax(i) + Bu(i), \forall i < k\}$;
$X_{ns}^p(k) = \bigcup_{x_0 \in X_{ns}^p(0)} \bigcup_{U^k} \{x : x(i+1) = Ax(i) + Bu(i), \forall i < k\}$
$CX_s^p(k) := \{y : y = Cx(k), x(k) \in X_s^p(k)\}$;
$CX_{ns}^p(k) := \{y : y = Cx(k), x(k) \in X_{ns}^p(k)\}$
if ($CX_s^p(k) \subseteq CX_{ns}^p(k)$) then
    $X_s^* = X_s^p(0)$;
    $X_{ns}^* = X_{ns}^p(0)$
else if ($CX_s^p(k) \cap CX_{ns}^p(k) = \emptyset$) then
    Run Algo. (1) with $X = X_s^p(0)$
    $X_s^* = X_s$;
    $X_{ns}^* = (X_s^p(0) \setminus X_s^*) \bigcup X_{ns}^p(0)$
else
    $X_s^{**}(0) := \{x \in X_s^p(0) : CX_s^{**}(k) \subseteq (CX_s^p(k) \cap CX_{ns}^p(k))\}$
    Run Algo. (1) with $X = X_s^p(0) \setminus X_s^{**}(0)$
    $X_s^* = X_s \bigcup X_s^{**}(0)$;
    $X_{ns}^* = (X_s^p(0) \setminus X_s^*) \bigcup X_{ns}^p(0)$
end if
```

Remark 8.2: We note that $k-$ISO can be achieved by a combination of considering a smaller set of secret states and a larger set of nonsecret states. The *else-if* and *else* statements in Algorithm (2) determine a 'compatible' pair such that X_s^* is $k-$ISO w.r.t. X_{ns}^*.

Proposition 8.3: Algorithm (2) is correct.

Proof: Three cases are considered.
I. $CX_s^p(k) \subseteq CX_{ns}^p(k)$: $k-$ISO, from Thm. (5.1).
II. $CX_s^p(k) \cap CX_{ns}^p(k) = \emptyset$: The proof, in this case, is identical to Lemma (8.1).
III. $CX_s^p(k) \cap CX_{ns}^p(k) \neq \emptyset$: for every $x_1 \in X_s^{**}(0)$, there exists $x_2 \in X_{ns}^p(0)$ such that $CX_s^{**}(k) \subseteq (CX_s^p(k) \cap CX_{ns}^p(k)) \subseteq CX_{ns}^p(k)$. Therefore, $X_s^{**}(0)$ is $k-$ISO with respect to $X_{ns}^p(0)$. From the proof of Lemma (8.1), X_s is $k-$ISO with respect to $X \setminus X_s = X_s^p(0) \setminus (X_s \cup X_s^{**}(0))$, which completes the proof. ■

IX. Conclusion and Future Work

We presented a framework for opacity in DT-LTI systems and established conditions to achieve $k-$ISO in terms of reachable sets. It was shown that unions of opaque sets preserve opacity, while this was not necessarily true for intersections. We then determined conditions under which $k-$ISO was equivalent to output controllability. Finally, we presented an algorithm to compute an opaque subset, given candidate sets of secret and nonsecret states, and proved its correctness.

The adversary not wanting to reveal itself was the motivation for allowing only snapshots of the output in our definition of $k-$ISO. Future work will involve examining the case when the adversary wishes to verify opacity while minimizing its number of observations and not revealing itself. We also propose to formalize a notion of $k-$ISO for distributed systems and CPS modeled as continuous time LTI systems and nonlinear systems. Tools from information theory [25] and differential privacy [26] can be used to develop a framework to quantify opacity in cyberphysical systems.

References

[1] J. Slay and M. Miller, *Lessons learned from the Maroochy water breach*. Springer, 2008.

[2] J. P. Farwell and R. Rohozinski, "Stuxnet and the future of cyber war," *Survival*, vol. 53, no. 1, pp. 23–40, 2011.

[3] A. A. Cárdenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems," in *HotSec*, 2008.

[4] S. Schneider and A. Sidiropoulos, "CSP and anonymity," in *Computer Security—ESORICS*. Springer, 1996, pp. 198–218.

[5] R. Focardi and R. Gorrieri, "A taxonomy of trace-based security properties for CCS," in *Proceedings of Computer Security Foundations Workshop*. IEEE, 1994, pp. 126–136.

[6] E. Badouel *et al.*, "Concurrent secrets," *Discrete Event Dynamic Systems*, vol. 17, no. 4, pp. 425–446, 2007.

[7] A. Saboori and C. N. Hadjicostis, "Notions of security and opacity in discrete event systems," in *46th IEEE Conference on Decision and Control*, 2007.

[8] J. Dubreil, P. Darondeau, and H. Marchand, "Supervisory control for opacity," *IEEE Trans. on Automatic Control*, vol. 55, no. 5, pp. 1089–1100, 2010.

[9] A. Saboori and C. N. Hadjicostis, "Opacity-enforcing supervisory strategies via state estimator constructions," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1155–1165, 2012.

[10] F. Pasqualetti, F. Dorfler, and F. Bullo, "Control-theoretic methods for cyberphysical security: Geometric principles for optimal cross-layer resilient control systems," *IEEE Control Systems*, vol. 35, no. 1, pp. 110–127, 2015.

[11] L. Mazaré, "Using unification for opacity properties," *Proceedings of the 4th IFIP WG1*, vol. 7, 2004.

[12] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM Journal on Control and Optimization*, vol. 25, no. 1, pp. 206–230, 1987.

[13] ——, "The control of discrete event systems," *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81–98, 1989.

[14] M. Sampath *et al.*, "Failure diagnosis using discrete-event models," *IEEE Transactions on Control Systems Technology*, vol. 4, no. 2, pp. 105–124, 1996.

[15] A. J. Van Der Schaft *et al.*, *An introduction to hybrid dynamical systems*. Springer London, 2000, vol. 251.

[16] R. R. Burridge *et al.*, "Sequential composition of dynamically dexterous robot behaviors," *The International Journal of Robotics Research*, vol. 18, no. 6, pp. 534–555, 1999.

[17] F. Cassez, J. Dubreil, and H. Marchand, "Synthesis of opaque systems with static and dynamic masks," *Formal Methods in System Design*, vol. 40, no. 1, pp. 88–115, 2012.

[18] A. Saboori and C. N. Hadjicostis, "Verification of initial-state opacity in security applications of DES," in *9th International Workshop on Discrete Event Systems*. IEEE, 2008, pp. 328–333.

[19] ——, "Verification of infinite-step opacity and complexity considerations," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1265–1269, 2012.

[20] F. Lin, "Opacity of DES and its applications," *Automatica*, vol. 47, no. 3, pp. 496–503, 2011.

[21] Y.-C. Wu and S. Lafortune, "Synthesis of insertion functions for enforcement of opacity security properties," *Automatica*, vol. 50, no. 5, pp. 1336–1348, 2014.

[22] ——, "Comparative analysis of related notions of opacity in centralized and coordinated architectures," *Discrete Event Dynamic Systems*, vol. 23, no. 3, pp. 307–339, 2013.

[23] A. Saboori and C. N. Hadjicostis, "Verification of k-step opacity and analysis of its complexity," *IEEE Transactions on Automation Science and Engineering*, vol. 8, no. 3, pp. 549–559, 2011.

[24] A. Van Der Schaft, "Equivalence of dynamical systems by bisimulation," *IEEE Transactions on Automatic Control*, vol. 49, no. 12, pp. 2160–2172, 2004.

[25] B. Bérard, J. Mullins, and M. Sassolas, "Quantifying opacity," *Mathematical Structures in Computer Science*, vol. 25, no. 02, pp. 361–403, 2015.

[26] C. Dwork, "Differential privacy," in *Encyclopedia of Cryptography and Security*. Springer, 2011, pp. 338–340.