Initial State Opacity in Stochastic DES

C. Keroglou  
ECE Department  
University of Cyprus  
ckerog01@ucy.ac.cy

C. N. Hadjicostis  
ECE Department  
University of Cyprus  
chadjie@ucy.ac.cy

Abstract

A non-deterministic finite automaton is initial-state opaque if the membership of its true initial state to a given set of secret states $S$ remains opaque (i.e., uncertain) to an intruder who observes system activity through some natural projection map. By establishing that the verification of initial state opacity is equivalent to the language containment problem, earlier work has established that the verification of initial state opacity is a PSPACE-complete problem. In this paper, motivated by the desire to incorporate probabilistic (likelihood) information, we extend the notion of initial state opacity to stochastic discrete event systems. Specifically, we consider systems that can be modeled as probabilistic finite automata, and introduce and analyze the notions of almost initial state opacity and step-based almost initial state opacity, both of which hinge on the a priori probability that the given system generate behavior that violates initial-state opacity. We also discuss how almost initial state opacity and step-based almost initial state opacity can be verified, and analyze the complexity of the proposed verification methods.

Keywords: Initial state opacity, probabilistic automata, security and privacy

1. Introduction

The increased reliance on shared cyber-infrastructures by many applications (ranging from defense and banking to health care and power distribution systems) has led to the study of various notions of security and privacy. One category of such notions focuses on the information flow from the system to the intruder [6, 16]. In particular, opacity is a security notion that aims to determine 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 [2, 9]. Typically, this requires that the intruder, who observes the system’s behavior through some observation mask (typically taken to be a natural projection map), never be able to establish the truth of the predicate.

The work in this paper is related to earlier work that studied notions of opacity in discrete event systems, such as [3] (which focuses on finite state Petri nets and defines opacity with respect to state-based predicates), [1] (which considers multiple intruders that are modeled as observers with different observation capabilities and requires that no intruder be able to determine that the actual trajectory of the system belongs to the secret language assigned to that intruder), and [5] (which considers a single intruder that might observe different events than the ones observed/controlled by the supervisor, and establishes that a minimally restrictive supervisor always exists, but might not be regular).

The work in [9] considered opacity with respect to state-based predicates in discrete event systems (DES) that can be modeled as non-deterministic finite automata with partial observation on their transitions. In particular, this line of work considered current state opacity [11], initial-state opacity [10], and $K$-step opacity [13], as well as pertinent applications in the tracking vehicles in wireless sensor environments [12], and the assessment of guarantees in anonymity protocols [8].

Assuming that the initial state of a given non-deterministic finite automaton $G$ is (partially) unknown, the notion of initial-state opacity (ISO) requires that no sequence of transitions allows the intruder (who has full knowledge of the system model and tracks the sequence of observable transitions) to unambiguously determine that the initial state of the system belonged to a given set of secret states $S$. The verification of initial-state opacity for a given non-deterministic finite automaton $G$ is polynomially equivalent to the language containment problem for a non-deterministic finite automaton [14] and, thus, it is a PSPACE-complete problem [7] (at least in its general form). One way to verify initial state opacity is to use the initial-state estimator for $G$, as discussed in [10] and reviewed in Section 2 of this paper; this straightforward method has complexity of $O(2^{N^2})$ where $N$ is the number of states of the given system $G$, but can be improved to $O(4^N)$ via the use of state-status mappings [15].

In this paper we extend ISO to stochastic systems. Specifically, we define and analyze the notions of step-based almost initial state opacity (SAISO) and almost initial-state opacity (AISO), which are inspired respectively by the notions of step-based almost current-state opacity and almost current-state opacity in [11]. Roughly
speaking, we can say that SAISO captures the a priori probability that the system will generate behavior that violates initial state opacity after a certain number of events, whereas AISI captures the a priori probability that the system will eventually generate behavior that violates the ISO property (with respect to a given set of secret initial states $S$). We compare these probabilities against a threshold $\theta$, $\theta > 0$, and if they are below the threshold we say that “the system is SAISO and/or AISI with respect to a given set of secret initial states $S$ and threshold $\theta$.” In order to verify SAISO and AISI, we use a combination of initial-state estimation techniques from [10] and Markov chain techniques (e.g., [11]).

2 Background on Initial State Estimation

2.1 Notation on Alphabets, Languages, and Relations

Let $\Sigma$ be an alphabet (set of events) and denote by $\Sigma^*$ the set of all finite-length strings of elements of $\Sigma$ (sequences of events), including the empty string $\epsilon$ (the length of a string $s$ is denoted by $|s|$ with $|\epsilon| = 0$). A language $L \subseteq \Sigma^*$ is a subset of finite-length strings (sequences of events with the first event appearing on the left) from strings in $\Sigma^*$ [4]. Given strings $s, t \in \Sigma^*$, the string $st$ denotes the concatenation of $s$ and $t$, i.e., the sequence of events captured by $s$ followed by the sequence of events captured by $t$. For a string $s$, $\overline{s}$ denotes the prefix-closure of $s$ and is defined as $\overline{s} = \{ t \in \Sigma^* \mid \exists t' \in \Sigma^* \{ tt' = s \} \}$.

A discrete event system (DES) is modeled in this paper as a non-deterministic finite automaton $G = (X, \Sigma, \delta, X_0)$, where $X = \{ 1, 2, \ldots, N \}$ is the set of states, $\Sigma$ is the set of events, $\delta : X \times \Sigma \rightarrow 2^X$ is the non-deterministic state transition function, and $X_0 \subseteq X$ is the set of possible initial states. For a set $Q \subseteq X$ and $\sigma \in \Sigma$, we define $\delta(Q, \sigma) = \cup_{q \in Q} \delta(q, \sigma)$; with this notation, at hand, the function $\delta$ can be extended from the domain $X \times \Sigma$ to the domain $X \times \Sigma^*$ in the routine recursive manner: $\delta(x, s\sigma) := \delta(x, \sigma) \delta(x, s)$ for $x \in X, s \in \Sigma^*$ and $\sigma \in \Sigma$ with $\delta(i, \epsilon) := \{ i \}$. The behavior of DES $G$ is captured by $L(G) := \{ s \in \Sigma^* \mid \exists x_0 \in X_0 \delta(x_0, s) \neq \emptyset \}$. We use $L(G, x)$ to denote the set of all traces that originate from state $x$ of $G$ (so that $L(G) = \bigcup_{x \in X_0} L(G, x_0)$).

In general, only a subset $\Sigma_{\text{obs}}$ (for $\Sigma_{\text{obs}} \subseteq \Sigma$) of the events can be observed, so that $\Sigma$ is partitioned into the set of observable events $\Sigma_{\text{obs}}$ and the set of unobservable events $\Sigma_{\text{unob}} = \Sigma - \Sigma_{\text{obs}}$. The natural projection $P_{\Sigma_{\text{obs}}} : \Sigma^* \rightarrow \Sigma_{\text{obs}}^*$ can be used to map any trace executed in the system to the sequence of observations associated with it. This projection is defined recursively as $P_{\Sigma_{\text{obs}}} (s) = P_{\Sigma_{\text{obs}}} (\sigma) P_{\Sigma_{\text{obs}}} (s)$, $\sigma \in \Sigma, s \in \Sigma^*$, with

$$P_{\Sigma_{\text{obs}}} (\sigma) = \begin{cases} \sigma, & \text{if } \sigma \in \Sigma_{\text{obs}}, \\ \epsilon, & \text{if } \sigma \in \Sigma_{\text{unob}} \cup \{ \epsilon \}, \end{cases}$$

where $\epsilon$ represents the empty trace [4]. In the sequel, the subscript $\Sigma_{\text{obs}}$ in $P_{\Sigma_{\text{obs}}}$ will be dropped when it is clear from context.

Any $m \in 2^X$ is a subset of $X^2$ (i.e., a relation on the set $X$) and contains pairs of states. In this paper, $m$ will be viewed as a state mapping consisting of pairs of a starting state and an ending state. The set of states included as the first (second) component in these pairs is called the set of starting (ending) states of $m$. We denote the set of starting states for state mapping $m$ by $m_1$, and the set of ending states by $m_2$. The composition operator $\circ : 2^{X^2} \times 2^{X^2} \rightarrow 2^{X^2}$ for state mappings $m_1, m_2 \in 2^{X^2}$ is defined as

$$m_1 \circ m_2 := \{ (i_1, i_3) \mid \exists i_2 \in X \{ (i_1, i_2) \in m_1, (i_2, i_3) \in m_2 \} \}.$$

We can map any observation of finite but arbitrary length in DES $G$ to a state mapping by using the mapping $M : \Sigma_{\text{obs}}^* \rightarrow 2^{X^2}$ defined for $\omega \in \Sigma_{\text{obs}}^*$ as

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

which we call the $\omega$-induced state mapping. The pair $(i, j) \in M(\omega)$ implies that there exists a sequence of events that starts from state $i$ and ends in state $j$, and produces observation $\omega$.

Finally, for any $Z \subseteq X$, we define the operator $\circ : 2^{X^2} \rightarrow 2^{X^2}$ to represent $Z \circ Z := \{ (i, i) \mid i \in Z \}$.

Theorem 1 Consider 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 $\Sigma_{\text{obs}}$, $\Sigma_{\text{obs}} \subseteq \Sigma$. Given two observation sequences $\omega_1 \in \Sigma_{\text{obs}}$ and $\omega_2 \in \Sigma_{\text{obs}}$, we have

$$M(\omega_1) \circ M(\omega_2) = M(\omega_1 \omega_2) \ .$$

Proof

We establish the following two directions:

1) If $(i, j) \in M(\omega_1) \circ M(\omega_2)$ then $(i, j) \in M(\omega_1 \omega_2)$: According to the definition of $\omega$-induced state mapping, a pair of states $(i, k) \in M(\omega_1)$ means that there exists a string $s_1$ such that (i) $P(s_1) = \omega_1$ and (ii) $k = \delta(i, s_1)$. Similarly, a pair of states $(k, j) \in M(\omega_2)$ means that there exists a string $s_2$ such that (i) $P(s_2) = \omega_2$ and (ii) $j = \delta(k, s_2)$. Note that, under the above conditions, $(i, j) \in M(\omega_1) \circ M(\omega_2)$. We also see that $(i, j) \in M(\omega_1 \omega_2)$: the string $s_1 \omega_2$ has projection $P(s_1 \omega_2) = \omega_1 \omega_2$ (follows from the definition of the natural projection) and satisfies $j = \delta(i, s_1 \omega_2)$ (follows from the definition of $\delta$); thus, $(i, j) \in M(\omega_1 \omega_2)$.

2) If $(i, j) \in M(\omega_1 \omega_2)$ then $(i, j) \in M(\omega_1) \circ M(\omega_2)$: According to the definition of $\omega$-induced state mapping, a pair of states $(i, j) \in M(\omega_1 \omega_2)$ means that there exists a string $s$ such that (i) $P(s) = \omega_1 \omega_2$ and (ii) $j = \delta(i, s)$. Clearly, we can find a prefix $s_1$ of $s$ such that $s_1 \omega_2 = s$ for some $s_2$, and $P(s_1) = \omega_1$ and $P(s_2) = \omega_2$. Pick a $k \in X$ such that $k = \delta(i, s_1)$ and $j = \delta(k, s_2)$ (such a $k$ exists because otherwise $j \notin \delta(i, s)$ which is a contradiction). Then, $(i, k) \in M(\omega_1)$ and $(k, j) \in M(\omega_2)$. We conclude that $(i, j) \in M(\omega_1) \circ M(\omega_2)$. 


2.2 Initial State Estimation and Initial-State Opacity

Definition 1 (Initial-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 \( \Sigma_{obs} \), \( \Sigma_{obs} \subseteq \Sigma \), the initial-state estimate after observing string \( \omega \in \Sigma_{obs}^* \) is defined as

\[
\hat{X}_0(\omega) = \{ x_0 \in X_0 \mid \exists s \in \Sigma^* \; \{ P(s) = \omega, \delta(x_0, s) \neq \emptyset \} \}.
\]

The authors of [10] introduced the construction of the initial-state estimator (ISE), i.e., a deterministic finite automaton \( G_{obs} = (X_{Iobs}, \Sigma_{obs}, \delta_{Iobs}, X_{Iobs0}) \) that is driven by observable events in \( \Sigma_{obs} \) and whose states are state mappings, i.e., \( X_{Iobs} \subseteq 2^X \). The ISE construction ensures the following property: given the observation of a sequence of labels \( \omega \in \Sigma_{obs}^*, \omega \neq \epsilon \) (generated by unknown underlying activity in the system \( G \)), the ISE reaches a state \( m = \delta_{Iobs}(X_{Iobs0}, \omega) \) such that the set of possible initial states is captured by the initial states associated with \( m \), i.e., for \( \omega \neq \epsilon \)

\[
\hat{X}_0(\omega) = m(1) \text{ where } m = \delta_{Iobs}(X_{Iobs0}, \omega).
\]

To construct the estimator \( G_{Iobs} \), we start from a state in which nothing about the initial system state is known; specifically, the state mapping associated with this initial state of the estimator is \( X_0 \circ X_0 \), where \( X_0 \) is the set of initial states of the system. The observation of a label \( \alpha \in \Sigma_{obs} \) causes \( G_{Iobs} \) to transition to the state associated with the state mapping obtained by composing the previous state mapping and the mapping \( M(\alpha) \) induced by the new observation. The information captured by this composed state mapping (and by each subsequently obtained state of \( G_{Iobs} \)) is the following: we keep track of all pairs of one starting state (from the set \( X_0 \)) and one ending state, such that we can reach the ending state from the starting state via a sequence of events that generates the sequence of observations seen so far. Note that this construction (which is described formally below) is guaranteed to be finite and has at most \( 2^{2^X} \) states, where \( N \) is the number of states of the finite automaton \( G \).

Definition 2 (Initial-State Estimator (ISE)) 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 \( \Sigma_{obs}, \Sigma_{obs} \subseteq \Sigma \), we define the initial-state estimator as the deterministic finite automaton \( G_{Iobs} = AC(2^{2^X}, \Sigma_{obs}, \delta_{Iobs}, X_{Iobs0}) \) with state set \( 2^{2^X} \) (power set of \( X \times X \)), event set \( \Sigma_{obs} \), initial state \( X_{Iobs0} = X_0 \circ X_0 \), and state transition function \( \delta_{Iobs} : 2^{2^X} \times \Sigma_{obs} \rightarrow 2^{2^X} \) defined for \( \alpha \in \Sigma_{obs} \) as

\[
m'(m, \alpha) := m \circ M(\alpha),
\]

where \( m, m' \in 2^{2^X} \). Recall that \( M(\alpha) \) denotes the state mapping that is induced by symbol \( \alpha \in \Sigma_{obs} \) and \( AC \) denotes the states that are accessible from initial state \( X_{Iobs0} \) via \( \delta_{Iobs} \). If we let \( X_{Iobs} \subseteq 2^{2^X} \) be the reachable states from the initial state \( X_{Iobs0} \) under \( \delta_{Iobs} \), then \( G_{Iobs} = (X_{Iobs}, \Sigma_{obs}, \delta_{Iobs}, X_{Iobs0}) \).

Adding a self-loop to each state of DFA \( G_{Iobs} \) for each label in the set \( \Sigma_{uo} = \Sigma - \Sigma_{obs} \) we create the DFA \( G_{Iobs} = (X_{Iobs}, \Sigma, \delta_{Iobs}, X_{Iobs0}) \), which is of use later in this paper. The formal definition of \( G \) can be found below.

Definition 3 (ISE with Unobservable Self-Loops) Consider 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 \( \Sigma_{obs}, \Sigma_{obs} \subseteq \Sigma \). Given its initial-state estimator \( G_{Iobs} = AC(2^{2^X}, \Sigma_{obs}, \delta_{Iobs}, X_{Iobs0}) \) \( \equiv (X_{Iobs}, \Sigma_{obs}, \delta_{Iobs}, X_{Iobs0}) \), we define the ISE with unobservable self loops to be the DFA \( G_{Iobs} = (X_{Iobs}, \Sigma, \hat{\delta}_{Iobs}, X_{Iobs0}) \) where we define \( \hat{\delta}_{Iobs} \) for \( x_{Iobs} \in X_{Iobs} \) and \( \sigma \in \Sigma \) as

\[
\hat{\delta}_{Iobs}(x_{Iobs}, \sigma) = \{ \delta(x_{Iobs}, \sigma), \text{ for } \sigma \in \Sigma_{obs}, \}
\]

\[
\text{for } \sigma \in \Sigma_{uo},
\]

where \( \Sigma_{uo} = \Sigma - \Sigma_{obs} \).

Example 1 The following example is used to clarify the notation and the ISE construction. Consider the finite automaton \( G = (X, \Sigma, \delta, X_0) \) shown on Fig. 1, where \( X = \{1, 2, 3, 4\}, \Sigma = \{\alpha, \beta\} \), \( \delta \) is as defined by the transitions in the figure, and \( X_0 = X = \{1, 2, 3, 4\} \). Assume that \( \Sigma_{obs} = \{\alpha, \beta\} \) and \( \Sigma_{uo} = \emptyset \). To construct the \( \alpha \)-induced state mapping, i.e., \( M(\alpha) \), note that \( \alpha \) can be observed from state 1. Upon this observation, if the initial state was 1, the ending state can be state 2 or state 3. Hence, \( M(\alpha) = \{(1, 2), (1, 3)\} \). Following the same reasoning, we obtain \( M(\beta) = \{(2, 3), (2, 4), (3, 3), (4, 1), (4, 3)\} \). The composition of these two state mappings yields \( M(\alpha) \circ M(\beta) = \{(1, 3), (1, 4)\} \) and indicates that if we observe \( \alpha \beta \), we could start from state 1 and end up in state 3 or state 4.

**Figure 1.** Non Deterministic Automaton (G) used in Example 1.

Fig. 2 shows the initial-state estimator for this system. The initial uncertainty is assumed to be equal to the state space and hence \( m_0 = X_0 \circ X_0 = \{(1, 1), (2, 2), (3, 3), (4, 4)\} \). (In Fig. 3 we use a graphical way to describe the pairs associated with each state
of the ISE. Upon observing $\alpha$, the next state of the ISE becomes
\[ m' = \delta_{Iobs}(m_0, \alpha) = m_0 \circ M(\alpha) = \{(1, 2), (1, 3)\} = M(\alpha) \equiv m_1. \]

Next, assume that we observe $\beta$; following the same reasoning as in the case of $M(\alpha)$, we first obtain $M(\beta) = \{(2, 4), (2, 3), (3, 3), (4, 1), (4, 3)\}$ and then we have
\[ m' = \delta_{Iobs}(m_1, \beta) = m_1 \circ M(\beta) = M(\alpha) \circ M(\beta) = \{(1, 3), (1, 4)\}. \]

Using this approach for all possible observations (from each state), the ISE construction can be completed as shown in Figs. 2 and 3. Note that in Figs. 2 and 3, following convention, we did not draw the state corresponding to the empty state mapping and transitions from/to it (e.g., from state $m_1$ under observation $\alpha$). The procedure is guaranteed to complete in finite time because the set of different state mappings that can be generated is finite.  

\[ \forall x_{Iobs} \in X_{Iobs} : x_{Iobs}(1) \not\subseteq S \text{ OR } x_{Iobs} = \emptyset. \]

Thus, one way to verify initial-state opacity is to first construct the ISE which has space complexity $O(2N^2)$ and similar time complexity, and check whether the above condition holds. For instance, due to the existence of state $m_4$ in the ISE in Fig. 2, the system $G$ in Fig. 1 is not initial state opaque with respect to the set of secret states $S = \{4\}$; however, the system is initial state opaque with respect to the set of secret states $S = \{3\}$. Note that one can reduce the complexity of the verification method to $O(4^N)$ via the use of state-status mappings [8, 15].

2.3 Probabilistic Finite Automata

A stochastic discrete event system (SDES) is modeled in this paper as a probabilistic finite automaton (PFA) $H = (X, \Sigma, p, \pi_0)$ where $X = \{1, 2, \ldots, N\}$ is the set of states, $\Sigma$ is the set of events, $\pi_0$ is the initial-state probability distribution vector, and $p(i', \sigma | i)$ is the state transition probability defined for $i, i' \in X$, and $\sigma \in \Sigma$, as the probability that event $\sigma$ occurs and the system transitions to state $i'$ given that the system is in state $i$. When $p(i', \sigma | i) = 0$, state $i'$ is not reachable from state $i$ via event $\sigma$ (in the diagram representing the given
PFA, we do not draw such transitions). Clearly, we have
\[ \sum_{i' \in X} \sum_{\sigma \in \Sigma} p(i', \sigma | i) = 1, \forall i \in X. \]

We can assign a probability to each trace in \( \Sigma^* \) with the interpretation that this value determines the probability of occurrence of this trace: if \( Pr(i', s) \) denotes the probability that \( s \) is executed in the system and the current state of the system becomes state \( i' \), then we can define for \( \sigma \in \Sigma, s \in \Sigma^* \), [8]

\[ Pr(s|\sigma) = \sum_{i \in X} Pr(i, \sigma | i) \]
\[ Pr(i, \sigma | i') = \sum_{i' \in X} p(i', \sigma | i) Pr(i', s) \]
\[ Pr(i, \epsilon) = \pi_0(i) \]  

(1)

Given a PFA \( H = (X, \Sigma, p, \pi_0) \) we can associate with it a unique NFA \( G = (X, \Sigma, \delta, X_0) \) where the state transition function \( \delta : X \times \Sigma \rightarrow 2^X \) is defined for \( i, i' \in X, \sigma \in \Sigma \) as

\[ i' = \delta(i, \sigma) \text{ if } p(i', \sigma | i) > 0, \]
and the set of possible initial states is defined as \( X_0 = \{ i | \pi_0(i) > 0 \} \). In this way, the behavior of the PFA \( H \) is mapped to the behavior of the associated NFA \( G \), i.e., \( L(H) = L(G) \) (where \( L(H) = \{ s \in \Sigma^* | Pr(s) > 0 \} \)).

3 Initial-State Opacity in Stochastic DES

Based on Definition 4, a given system is not initial-state opaque even if the probability of observing a sequence of observations that reveals that the system initial state is within a set of secret states is very small. One way to quantify initial-state opacity is to obtain the \textit{a priori} probability with which the system might generate behavior that violates initial-state opacity. In order to calculate this probability, we take advantage of the fact that if the set of initial states is exposed to be within the given set of secret states \( S \) for a given sequence of observations, then it remains exposed for all possible continuations of that sequence.

Definition 5 below characterizes the probability of violating initial-state opacity assuming that the system has generated \( k \) events; this is referred to as step-based almost initial-state opacity (SAISO). Definition 6 is the extension of SAISO when there is no consideration regarding the length of the sequence (i.e., it considers the probability of violating initial-state opacity following any sequence of events), and requires that this probability lies below a threshold. This notion of opacity is referred to as almost initial-state opacity (AISO). Both SAISO and AISO are inspired, respectively, by step-based almost current-state opacity and almost current-state opacity in [11]. The key difference is that in the case of current state opacity, there is monotonicity (a violation of current state opacity at time step \( k \) does not imply a violation of current state opacity at later times) which renders the two definitions — in the case of current state opacity — quite distinct.

**Definition 5 (Step-Based Almost Initial-State Opacity)**

Given a PFA \( H = (X, \Sigma, p, \pi_0) \) and its associated NFA \( G = (X, \Sigma, \delta, X_0) \), a natural projection map \( P \) with respect to the set of observable events \( \Sigma_{obs}, \Sigma_{obs} \subseteq \Sigma \), and a set of secret states \( S \subseteq X_0 \), we define

\[ L_I = \{ s \in L(G) | \hat{X}_0(P(s)) \subseteq S \} \]

(where \( \hat{X}_0(\omega) \) is the set of initial state estimates following the observation sequence \( \omega \)). Then, PFA \( H \) is step-based almost initial-state opaque (SAISO) with respect to \( S, P \), and a threshold \( \theta \) (or \( (S, P, \theta, T) \)-almost initial-state opaque) if

\[ \sum_{s \in L_I, |s|=k} Pr(s) < \theta, \forall k = 0, 1, 2, \ldots \]

**Remark 1** Note that the language \( L_I \) in Definition 5 denotes the set of strings \( s \) in the system that violate initial-state opacity (ISO). Note that if \( G \) is initial-state opaque, then \( H \) is \( (S, P, \theta, T) \)-almost initial-state opaque for any \( \theta > 0 \).

**Definition 6 (Almost Initial-State Opacity)**

Given a PFA \( H = (X, \Sigma, p, \pi_0) \) and its associated NFA \( G = (X, \Sigma, \delta, X_0) \), a natural projection map \( P \) with respect to the set of observable events \( \Sigma_{obs}, \Sigma_{obs} \subseteq \Sigma \), and a set of secret states \( S \subseteq X_0 \), we define

\[ L_I = \{ s \in L(G) | \hat{X}_0(P(s)) \subseteq S \}, \]
\[ L^\theta_I = \{ s \in L_I | \forall s' \in \pi, s' \neq s' \notin L_I \}. \]

Then, PFA \( H \) is almost initial-state opaque (AISO) with respect to \( S, P \), and a threshold \( \theta \) (or \( (S, P, \theta) \)-almost initial-state opaque) if

\[ \sum_{s \in L^\theta_I} Pr(s) < \theta. \]

**Remark 2** If an observation sequence \( s \) causes a violation of initial-state opacity, then any \( s' \in L/s \) (any continuation of \( s \)) will also cause a violation of initial-state opacity. This can be easily shown using the monotonic (non-increasing) property of initial state estimation, i.e., the fact that

\[ \hat{X}_0(\omega_1 \omega_2) \subseteq \hat{X}_0(\omega_1) \]

for all \( \omega_1, \omega_2 \in \Sigma_{obs}^* \). Due to this fact, it is not hard to establish that

\[ \sum_{s \in L_I, |s|=k} Pr(s) \leq \sum_{s \in L_I, |s|=k+1} Pr(s), k = 0, 1, 2, \ldots \]

and

\[ \sum_{s \in L_I} Pr(s) \leq \sum_{s \in L_I, |s|\rightarrow \infty} Pr(s). \]
3.1 Motivating Example: Tracking a Vehicle in a Grid

As motivation for studying the notion of stochastic initial state opacity, we present an example that was first discussed in [8]. Consider a vehicle capable of moving on a grid, such as the toy $2 \times 2$ grid in Fig. 4(a). If we use the cell number to denote the state of the vehicle, then the trajectory that the vehicle follows corresponds to a sequence of states and the origin of the trajectory is captured by the initial state of the vehicle. The vehicle possible movements are available via a kinematic model, i.e., a finite automaton whose states are associated with the state (cell) of the vehicle and whose transitions correspond to the movements of the vehicle that are allowed at each position (up, down, left, right, diagonal, etc. — the allowed movements will presumably depend on the underlying terrain that the grid is capturing). Fig. 4(b) depicts an example of a kinematic model for the vehicle that moves in the toy grid of Fig. 4(a).

Suppose sensors are deployed in the grid such that each sensor detects the presence of the vehicle in a cell or in some aggregation of cells; when the vehicle passes through a cell within the coverage of a sensor, this sensor emits a signal that indicates this event. To capture this information, we can enhance the kinematic model by assigning label $\alpha$ to all transitions that end in a cell within the coverage area of sensor $\alpha$. Since sensor coverage 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 sensors covering that location. Fig. 1 depicts the (non-deterministic) finite automaton $G$ that models both the kinematic model of the vehicle and the corresponding sensor readings for a particular set of sensor coverages; note that unobservable transitions correspond to locations that are not covered by any sensor. Essentially $G$ is a non-deterministic finite automaton with partial observation on its transitions.

We can extend the above formulation to a probabilistic setting under which each transition is assigned a specific probability of occurrence as in Fig. 5. In this probabilistic setting we are interested in the probability of occurrence of violating strings (strings that resolve the initial state of the system).

Several security and privacy questions pertaining to the trajectory that the vehicle follows can be formulated in terms of state-based notions of opacity for automaton $G$ [8]. In particular, one of the questions that might arise in the above context is that of understanding whether the sensory information that is available allows us to obtain important information about the origin (initial state) of the vehicle. In a probabilistic setting this translates to the requirement that the probability of violating strings be under or above a specific threshold. For example, if we want maximum information about the initial state, we might want to strategically design the sensor network (in order to maximize the probability to capture the kinematic information). This task is trivial if sensors can be placed at each of the strategic locations; however, it becomes more challenging if there are constraints on the locations of the sensors.

3.2 Verifying Step-Based Almost Initial-State Opacity (SAISO)

In order to verify SAISO for a given PFA $H$, we need to obtain, for $k = 0, 1, 2, \ldots$, the cumulative probability of violating ISO for all strings of length $k$. The sequences of observations that violate ISO are captured in the ISE for the NFA $G$ associated with the PFA $H$ by the sequences of observations $(\omega)$ that reach a state in the ISE whose associated initial state estimates fall within a subset of secret states (i.e., $X_0(\omega) \subseteq S$).

Suppose we reach such a state via a string $s$ that generates a sequence of observations $\omega = P(s)$ such that $X_0(\omega) \subseteq S$; then all possible continuations of string $s$, will also violate ISO (this follows from the fact that $X_0(\omega_1 \omega_2) \subseteq X_0(\omega_1)$ discussed in Remark 2). We need to compute the a priori probability of all strings of length $k$, $k = 0, 1, 2, \ldots$, that violate ISO. In order to do this, we can introduce a product operator similar to the product operator for two NFA, which acts on a PFA and an NFA such that the probabilities associated with events in the PFA are retained. Then, the labels on the transitions can be discarded and we can compute the probability of violating ISO by constructing an appropriate Markov chain. The details of this construction are provided below.

Given a PFA $H = (X, \Sigma, p, \pi_0)$ and its associated NFA $G = (X, \Sigma, \delta, X_0)$, a natural projection map $P$ with respect to the set of observable events $\Sigma_{\text{obs}}$, $\Sigma_{\text{obs}} \subseteq \Sigma$, and a set of secret states $S \subseteq X$, we can verify AISO as follows:

1. Construct the (deterministic) Initial State Estimator (ISE), $G_{1\text{obs}} = (X_{1\text{obs}}, \Sigma_{\text{obs}}, \delta_{1\text{obs}}, X_{1\text{obs}})$ associated with $G$ and natural projection map $P$.
2. Construct $G_{1\text{obs}} = (X_{1\text{obs}}, \Sigma, \delta_{1\text{obs}}, X_{1\text{obs}})$ by inserting to $G_{\text{obs}}$ self-loops associated with each label in $\Sigma_{\text{uno}} \equiv \Sigma - \Sigma_{\text{obs}}$.
3. Construct PFA $H_p = H \times \hat{G}_{1\text{obs}} :=$
\((X_p, \Sigma, p_p, \pi_0, p)\) where \(X_p = X \times X_{\text{obs}}\) is the set of states, \(p_p(i'_p, \sigma|i_p)\) is the state transition probability defined for \(i_p = (i, I) \in X_p\) (i.e., \(i \in X\) and \(I \in X_{\text{obs}}\), \(i'_p = (i', I) \in X_p\), and \(\sigma \in \Sigma\), as \(p_p(i'_p, \sigma|i_p) = p(i', \sigma|i)\), if \(I' = \delta_{\text{obs}}(I, \sigma)\), and \(p_p(i'_p, \sigma|i_p) = 0\), otherwise; \(\pi_0, p\) is the initial-state probability distribution vector.

**Step 4:** Construct the Markov chain \(M = (X_p, p, \pi_0, p)\) associated with the PFA \(H_p = (X_p, \Sigma, p, \pi_0)\), i.e., obtain the Markov chain with the state transition probabilities \(p = (i'_p, \sigma|i_p)\) for \(i, i'_p \in X_p\) as \(p = p_M(i'_p|i_p) = \sum_{\sigma \in \Sigma} p_M(i'_p, \sigma|i_p)\).

**Example 2** (continued) Consider the PFA \(H\) depicted in Fig. 5 with set of secret states \(S = \{4\}\) and \(\pi_0 = [0.25, 0.25, 0.25, 0.25]''\) (i.e., each state is equally likely at the initialization of the system). To obtain the Markov chain \(M\) that can be used for verifying \(\text{AISO}\), we first construct the Initial State Estimator (ISE) \(G_{\text{obs}}\) for the NFA that corresponds to \(H\) (Step 1). Notice that the NFA associated with the PFA in Fig. 5 is the system \(G\) shown in Fig. 1 and its ISE has already been presented in Figs. 2 and 3. Since in this case we have \(\Sigma_{\text{obs}} = \Sigma\), the ISE with unobservable self loops in Step 2 is identical to the ISE (i.e., \(G_{\text{obs}} = G_{\text{obs}}\)).

Before completing Steps 3 and 4, we make the following simplification. We only need the probability of strings that violate ISO and not the probability of violating string at specific states of the ISE; moreover, we know that if an ISE state is associated with an initial state estimate that is a subset of \(S\) (or of \(X - S\)), then any continuation from this state will also lead to a state associated with an initial state estimate that is a subset of \(S\) (or of \(X - S\)). Thus, we can simplify the ISE construction by making certain states of the ISE absorbing. For this example, starting with the ISE in Fig. 2, we can obtain the reduced ISE shown on the left of Fig. 6.

A simplified version of PFA \(H_p\) (Step 3) is depicted on the right of Fig. 6. In this figure we draw only the accessible states when we start from the secret state 4 (we can do this because the initial state estimates always include the true initial state, and the only secret state in the example is \(\{4\}\)). The associated transition probabilities are as depicted in Fig. 7. Finally, the underlying Markov chain \(M\) (Step 4) is easily obtained by discarding the labels of transitions on PFA \(H_p\).

In order to verify \(\text{SAISO}\), we need to verify that the total probability of violating strings is always lower than a threshold \(\theta > 0\). Given the underlying Markov chain, we need to compute the probability of all strings that reach a state, whose associated initial state estimate is a subset of the set of secret states, at \(k\) step, for each \(k \in \{0, 1, 2, \ldots\}\). In other words, we need to compute the a priori probability that we will be at a violating state (i.e., a state in which the associated initial state estimate is a subset of the set of secret states \(S\)), following a sequence of \(k\) events in the system.

**4 Conclusions**

In this paper we introduced and characterized the notions of step-based almost initial-state opacity (SAISO) and almost initial-state opacity (AISO) in stochastic DES, as extensions of the notion of initial-state opacity in non-deterministic DES. The notion of initial-state opacity requires that the membership of the initial state of a given NFA to a set of secret states remain opaque to intruders (i.e., an intruder can never be certain that the system originated from a state within the set of secret states). We define almost initial-state opacity (AISO) by requiring the
"a priori" probability that behavior in the system will violate initial state opacity be below a threshold. Similarly, we define step-based almost initial-state opacity (SAISO) by requiring that, following a sequence of \(k\) events, the "a priori" probability that the system will violate initial state opacity is below a given threshold for each \(k = 0, 1, 2, ...\).

In the future, we are interested in connecting the problem of almost initial-state opacity to that of classification between two hidden Markov models (HMMs).

5 Acknowledgments

This work falls under the Cyprus Research Promotion Foundation (CRPF) Framework Programme for Research, Technological Development and Innovation 2009–2010 (CRPF’s FP 2009–2010), co-funded by the Republic of Cyprus and the European Regional Development Fund, and specifically under Grant TITIE/OPIZO/0609(BE)/08. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of CRPF.

References


