Quantification of Secrecy in Partially Observed Stochastic Discrete Event Systems

Jun Chen, *Member, IEEE*, Mariam Ibrahim, *Member, IEEE*, and Ratnesh Kumar, *Fellow, IEEE*

***Abstract*— While cryptography is used to protect the content of information (e.g., a message) by making it undecipherable, behaviors (as opposed to information) may not be encrypted and may only be protected by partially or fully hiding through creation of ambiguity (by providing covers that generate indistinguishable observations from secrets). Having a cover together with partial observability does cause ambiguity about the system behaviors desired to be kept secret, yet some information about secrets may still be leaked due to statistical difference between the occurrence probabilities of the secrets and their covers. In this paper, we propose a Jensen–Shannon divergence (JSD)-based measure to quantify secrecy loss in systems modeled as partially observed stochastic discrete event systems, which quantifies the statistical difference between two distributions, one over the observations generated by secret and the other over those generated by cover. We further show that the proposed JSD measure for secrecy loss is equivalent to the mutual information between the distributions over possible observations and that over possible system status (secret versus cover). Since an adversary is likely to discriminate more if he/she observes for a longer period, our goal is to evaluate the worst case loss of secrecy as obtained in the limit over longer and longer observations. Computation for the proposed measure is also presented. Illustrative examples, including the one with side-channel attack, are provided to demonstrate the proposed computation approach.**

***Note to Practitioners*—Secrecy is the ability to hide private information. For *communicated information*, this can be done through encryption or access control. However, the same is not possible for system *behaviors*, and in contrast, cover is introduced for providing ambiguity. Quantifying the ability to hide secrets is a challenge. This paper provides a means to quantify this in terms of a type of distance measure between a secret and its cover. A computation of the same is also provided for partially observed stochastic discrete event systems and illustrated through a cache's side-channel secrecy loss example.**

Manuscript received July 19, 2015; revised October 19, 2015, March 13, 2016, and August 12, 2016; accepted August 19, 2016. Date of publication September 20, 2016; date of current version January 4, 2017. This paper was recommended for publication by Associate Editor C. N. Hadjicostis and Editor S. Reveliotis upon evaluation of the reviewers' comments. This work was supported in part by PNNL and John Deere through NSF-IUCRC, Security and Software Engineering Research Center and in part by the National Science Foundation under Grant NSF-CCF-1331390 and Grant NSF-ECCS-1509420. (Jun Chen and Mariam Ibrahim contributed equally to this work.)

J. Chen is with the Idaho National Laboratory, Idaho Falls, ID 83415 USA (e-mail: jun.chen@inl.gov).

M. Ibrahim is with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA, and also with the Department of Mechatronics Engineering, German Jordanian University, Amman 11180, Jordan (e-mail: mariami@iastate.edu).

R. Kumar is with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: rkumar@iastate.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2016.2604222

***Index Terms*— Discrete event systems (DESs), Jensen–Shannon divergence (JSD), partial observability, secrecy quantification, stochastic systems.**

I. Introduction

THE rapid progress in information and communication technology has made it possible for an adversary to eavesdrop and/or attack confidential or private communication. While cryptography is used to protect the content of information (e.g., a message) by making it undecipherable, the same technique may not be used to hide behaviors, which may not be encrypted. In such cases, *secrecy* can instead be attained through creation of ambiguity, caused, for example, by partial observation that ambiguates secrets from covers, where the secrets are system behaviors desired to be kept confidential, whereas the covers are the complementary system behaviors that generate the same observations as the secrets, creating ambiguity. Researchers in the field of security and privacy have explored many techniques for hiding secrets based on ambiguation schemes such as *steganography and watermarking* [1], [2], *network-level anonymization* [3], and *software obfuscation* [4].

Various notions of information secrecy have been explored in the literature. References [5]–[7] defined the noninterference for input–output systems as a property in which the outputs that are observable to an *adversary* should not depend on any *secret* input so that the adversary does not deduce anything about the secret input by observing the output. Noninterference is a logical notion that is either satisfied or violated, and as such, it does not allow the quantification of the degree to which a system may violate the property. Accordingly, the notion is enriched for probabilistic systems for which the degree of interference can be quantified in terms of the amount of information leaked by a system to an observer. The amount of information leakage, in turn, is measured by the loss of uncertainty about the inputs due to the observation of the outputs, i.e., the difference between the prior and posterior entropies of the inputs, namely, the mutual information between inputs and outputs [5]. While such a quantification of information leakage is satisfactory for long periods of system operation (since entropy measures uncertainty in an average sense), it is of limited use for systems in which an adversary makes a single observation. To address this situation, the average case measure of entropy was replaced by its best case measure, corresponding to minimum uncertainty, namely, *min-entropy*, in the definition of mutual information [7].

In general, a secret can be defined as a sequence of executions instead of a single execution, and this general situation has also been examined in the literature. For example, in the setting of discrete event systems (DESs), the definition of secrecy examined in [8] requires that the execution of behaviors constituting a secret must be masked to an observer through indistinguishable behaviors that are nonsecret (i.e., cover). This is indeed analogous to the notion of noninterference, which by virtue of being logical has the same limitation that it cannot quantify the degree to which a system is interfering (or leaks information).

For probabilistic DESs, where each discrete transition is associated with a certain occurrence probability, more powerful notions of secrecy can be defined. For example, [9] used Jensen–Shannon divergence (JSD) between the distributions of a secret versus its cover as a way to quantify the secrecy, which is measured as the divergence of two distributions over the set of feasible observations, one being the probabilities of secret behaviors and the other being those of cover behaviors. An approximation algorithm for computing an upper bound of JSD was also provided in [9]. Another attempt to generalize secrecy from logical to stochastic DESs is provided in the work of Saboori and Hadjicostis [10], where similar to the setting of mutual-information-based characterization of information leakage, they consider the difference between the prior and posterior distributions (before and after any observations) of the secret states and require it to be upper bounded. The corresponding verification problem turns out to be undecidable. Saboori and Hadjicostis [11] proposed the notion of *step-based almost current-state opacity* requiring that the probability of revealing the secret must be upper bounded at each time step. This notion is decidable, but stringent since it is defined for each individual step. In contrast, the definition of S_τ secrecy proposed by [12] bounds the probability of revealing the secret over the set of *all* behaviors, as opposed to that for each step. It is shown that S_τ-secrecy can be viewed as a generalization of the logical secrecy defined in [8] and that it is a variant of the divergence used in [9]. Related works on K-step and infinite step opacity are explored in [13] and [14]. The above-mentioned secrecy notions (also referred to *opacity* in the literature) along with the related articles have been reviewed in a recent survey [15].

In this paper, we propose a JSD-based quantification to measure the secrecy loss in partially observed stochastic DESs (SPODESs), which are Markovian generators of *arbitrary* long sequences with transitions being *partially observed*. (SPODESs are equivalent to partially observed labeled Markov chains; see the following text.) The proposed JSD-based quantification for secrecy loss is shown to be equivalent to the mutual information between the distributions over possible observations and that over possible status of system execution (secret versus cover). A *recursive* method for JSD computation is also presented: given the distribution with respect to length-$(n-1)$ sequences and the length-1 dynamics of the underlying partially observed model, it computes the JSD of length-(n) sequences. Under certain ergodicity conditions (see Section III-B), this recursion reaches a fixed point and provides "limiting" JSD measure, quantifying the worst case statistical difference, which is defined over arbitrary long sequences. Since JSD is always bounded between 1 and 0, this worst case value is also bounded, providing an upper bound to the quantification of the amount of information leaked about secrets due to statistical difference between the observations of secrets versus covers (see Remark 3). We derive the above recursion and then construct an observer model of the given SPODES, which is then used to develop a state-based computation of the fixed-point JSD measure.

The computation of JSD for a SPODES is challenging since a finite-state DES under partial observations is potentially infinite state (with the state space being conditional state distributions following observations). However, a finite-state stochastic observer can be constructed for computing conditional state distribution following observations, which we construct and employ for divergence computation. Note that such a stochastic observer was first introduced in [16] for analyzing diagnosability of SPODES. This observer model is not a Markov chain model since the transition probabilities are no longer scalars, rather matrices (not necessarily square). Secrecy has also been studied in the context of partially observed labeled Markov chain (POLMC). The reader is referred to [15] and [17] and the reference therein for the work on secrecy in the context of POLMC. While there is a slight difference between SPODES and POLMC (in a POLMC, transitions are not labeled and the states are partially observed, whereas in SPODES, transitions are labeled and they are partially observed), this does not affect the expressibility, as partial observability of states can be expressed as partial observability of transitions and vice versa. Note that [9] considers JSD over all finite length of observations for a terminating SPODES, while this paper formulates the limiting JSD for nonterminating SPODES, where the limiting JSD provides an upper bound to the level of loss of secrecy that is achieved when an intruder is able to wait for arbitrary long observations.

The rest of this paper is organized as follows. Section II presents notations and preliminaries. Divergence-based quantification of secrecy loss is provided in Section III, whereas Section IV presents an observer-based computation of worst case JSD resulting from arbitrary long observations. Section V illustrates the proposed approach through practical examples, while this paper is concluded in Section VI. The Appendix includes the proofs of lemmas and theorems.

II. Notations and Preliminaries

A. Stochastic PODESs

For an event set Σ, define $\bar{\Sigma} := \Sigma \cup \{\epsilon\}$, where ϵ denotes "no-event." The set of all finite-length event sequences over Σ, including ϵ, is denoted by Σ^*, and $\Sigma^+ := \Sigma^* - \{\epsilon\}$. A *trace* is a member of Σ^* and a *language* is a subset of Σ^*. We use $s \leq t$ to denote if $s \in \Sigma^*$ is a prefix of $t \in \Sigma^*$ and $|s|$ to denote the length of s or the number of events in s. For $L \subseteq \Sigma^*$, its prefix closure is defined as $pr(L) := \{s \in \Sigma^* | \exists t \in \Sigma^* : st \in L\}$ and L is said to be prefix closed (or simply closed) if $pr(L) = L$, i.e., whenever L contains a trace, it also contains all the prefixes of that trace.

A SPODES can be modeled by a *stochastic automaton* $G = (X, \Sigma, \alpha, x_0)$, where X is the set of states, Σ is the finite

set of events, $x_0 \in X$ is the initial state, and $\alpha : X \times \Sigma \times X \to [0, 1]$ is the probability transition function [18], and $\forall x \in X$, $\sum_{\sigma\in\Sigma}\sum_{x'\in X}\alpha(x, \sigma, x') = 1$. A non-stochastic partially observed discrete event system can be modeled as the same four-tuple, but by replacing the transition function with $\alpha : X \times \Sigma \times X \to \{0, 1\}$, and a non-stochastic partially observed discrete event system is deterministic if $\forall x \in X, \sigma \in \Sigma, \sum_{x'\in X}\alpha(x, \sigma, x') \in \{0, 1\}$. The transition probability function α can be generalized to $\alpha : X \times \Sigma^* \times X$ in a natural way. Define the language generated by G as $L(G) := \{s \in \Sigma^* \mid \exists x \in X, \alpha(x_0, s, x) > 0\}$. For a given G, a *component* $C = (X_C, \alpha_C)$ of G is a "subgraph" of G, i.e., $X_C \subseteq X$ and $\forall x, x' \in X_C$ and $\sigma \in \Sigma$, $\alpha_C(x, \sigma, x') = \alpha(x, \sigma, x')$ whenever the latter is positive, and $\alpha_C(x, \sigma, x') = 0$ otherwise. C is said to be a *strongly connected component* (SCC) or *irreducible* if $\forall x, x' \in X_C$, $\exists s \in \Sigma^*$ such that $\alpha_C(x, s, x') > 0$. An SCC C is said to be *closed* if for each $x \in X_C$, $\sum_{\sigma\in\Sigma}\sum_{x'\in X_C}\alpha_C(x, \sigma, x') = 1$. The states that belong to a closed SCC are *recurrent states* and the remaining states (which do not belong to any closed SCC) are *transient states*. Another way to identify recurrent versus transient states is to consider the steady-state state distribution π^* as the fixed point of $\pi^* = \pi^*\Omega$, where π^* is a row vector with the same size as X and Ω is the transition matrix with ijth entry being the transition probability $\sum_{\sigma\in\Sigma}\alpha(i, \sigma, j)$. (In case Ω is periodic with period $d \neq 1$, we consider the set of fixed points of $\pi^* = \pi^*\Omega^d$.) Then any state i is recurrent if and only if there exists a reachable fixed point π^* such that the ith entry of π^* is nonzero. Identifying the set of recurrent states can be done polynomially by the algorithm presented in [19].

The events executed by a SPODES can be partially observed by an observer (i.e., an adversary). Such limited observation capability of an observer can be represented as an observation mask, $M : \bar{\Sigma} \to \bar{\Delta}$, where Δ is the set of observed symbols and $M(\epsilon) = \epsilon$. An event σ is unobservable if $M(\sigma) = \epsilon$. The set of unobservable events is denoted by Σ_{uo} and the set of observable events is then given by $\Sigma - \Sigma_{uo}$. The observation mask can be generalized in a natural way to Σ^* with $M(\epsilon) = \epsilon$ and $\forall s \in \Sigma^*, \sigma \in \bar{\Sigma}, M(s\sigma) = M(s)M(\sigma)$.

B. Secret/Nonsecret Behaviors and Refined Plant

Suppose $K \subseteq \Sigma^*$ models the secret behaviors (traces), whereas the remaining traces in $L - K$ can be viewed as its cover. Let the stochastic automaton $G = (X, \Sigma, \alpha, x_0)$ with generated language $L(G) = L$ be the system model and the *deterministic* automaton $R = (Y, \Sigma, \beta, y_0)$, which specifies the secret behaviors K, be such that $L(R) = K$. Then a refinement of G with respect to R, denoted G^R, can be used to capture the property-satisfying/violating traces in form of the reachability of certain nonsecret states (the state has D in its second coordinate), and is given by $G^R := (X \times \bar{Y}, \Sigma, \gamma, (x_0, y_0))$, where $\bar{Y} = Y \cup \{D\}$ and $\forall (x, \bar{y}), (x', \bar{y}') \in X \times \bar{Y}, \sigma \in \Sigma, \gamma((x, \bar{y}), \sigma, (x', \bar{y}')) = \alpha(x, \sigma, x')$, if the following holds:

$$(\bar{y}, \bar{y}' \in Y \wedge \beta(\bar{y}, \sigma, \bar{y}') > 0) \vee (\bar{y} = \bar{y}' = D) \vee \left(\bar{y}' = D \wedge \sum_{y\in Y}\beta(\bar{y}, \sigma, y) = 0\right)$$

and otherwise $\gamma((x, \bar{y}), \sigma, (x', \bar{y}')) = 0$. Note that here D is an added state to capture the traces in $L - K$. Then it can be seen that the refined plant G^R has the following properties.

1) $L(G^R) = L(G)$.
2) Any property-satisfying trace $s \in L(G)$ but not in $L(R)$ transitions the refinement G^R to a nonsecret state.
3) For each $s \in L(G) = L(G^R)$, $\sum_{x\in X}\alpha(x_0, s, x) = \sum_{(x,\bar{y})\in X\times\bar{Y}}\gamma((x_0, y_0), s, (x, \bar{y}))$, i.e., the occurrence probability of each trace in G^R is the same as that in G.

For $(x, \bar{y}), (x', \bar{y}') \in X \times \bar{Y}$ and $\delta \in \Delta$ define the set of traces originating at $(x, \bar{y})$, terminating at $(x', \bar{y}')$, and executing a sequence of unobservable events followed by a single observable event with observation δ as $L_{G^R}((x, \bar{y}), \delta, (x', \bar{y}')) := \{s \in \Sigma^* | s = u\sigma, M(u) = \epsilon, M(\sigma) = \delta, \gamma((x, \bar{y}), s, (x', \bar{y}')) > 0\}$. Define $\alpha(L_{G^R}((x, \bar{y}), \delta, (x', \bar{y}'))) := \sum_{s\in L_{G^R}((x,\bar{y}),\delta,(x',\bar{y}'))}\gamma((x, \bar{y}), s, (x', \bar{y}'))$ and denote it by $\theta_{(x,\bar{y}),\delta,(x',\bar{y}')}$. Therefore, $\theta_{(x,\bar{y}),\delta,(x',\bar{y}')}$ is the probability of all traces originating at $(x, \bar{y})$, terminating at $(x', \bar{y}')$, and executing a sequence of unobservable events followed by a single observable event with observation δ. Also for $i = (x, \bar{y})$ and $j = (x', \bar{y}')$, define $\lambda_{ij} = \sum_{\sigma\in\Sigma_{uo}}\gamma(i, \sigma, j)$ as the probability of transitioning from $(x, \bar{y})$ to $(x', \bar{y}')$ while executing a single unobservable event. Then $\theta_{i,\delta,j} = \sum_k \lambda_{ik}\theta_{k,\delta,j} + \sum_{\sigma\in\Sigma:M(\sigma)=\delta}\gamma(i, \sigma, j)$, where the first term on the right-hand side (RHS) corresponds to transitioning in at least two steps (i to intermediate k unobservably and k to j with a single observation δ at the end), whereas the second term on the RHS corresponds to transitioning in exactly one step [20], [21]. Thus, for each $\delta \in \Delta$, all the probabilities $\{\theta_{i,\delta,j} | i, j \in X \times \bar{Y}\}$ can be found by solving the following matrix equation (see [22]–[24] for a similar equation):

$$\Theta(\delta) = \Lambda\Theta(\delta) + \Gamma(\delta) \tag{1}$$

where $\Theta(\delta)$, Λ, and $\Gamma(\delta)$ are all $|X \times \bar{Y}| \times |X \times \bar{Y}|$ square matrices whose ijth elements are given by $\theta_{i,\delta,j}$, λ_{ij} and $\sum_{\sigma\in\Sigma:M(\sigma)=\delta}\gamma((x, \bar{y}), \sigma, (x', \bar{y}'))$, respectively.

Remark 1: To find $\Theta(\delta)$ using (1), we need to solve $\Theta(\delta) = (I - \Lambda)^{-1}\Gamma(\delta)$. The complexity of matrix inverse is $O(|X|^3 \times |\bar{Y}|^3)$ and the complexity of matrix multiplication is $O(|X|^3 \times |\bar{Y}|^3)$, and so overall complexity is $O(|X|^3 \times |\bar{Y}|^3)$. Since the number of secret states and the number of nonsecret states are both upper bounded by the number of states in G^R, which is $O(|X| \times |\bar{Y}|)$, the complexity of finding $\Theta(\delta)$ for all $\delta \in \Delta$ using (1) is bounded by $O(|\Delta| \times |X|^3 \times |\bar{Y}|^3)$.

Note also G^R has $O(|X| \times |\bar{Y}|)$ states and $O(|\Sigma| \times |X|)$ transitions per state since only the G part is nondeterministic, whereas the complexity of identifying all the nonsecret recurrent states in G^R is cubic in the number of states in G^R and linear in the number of transitions in G^R, respectively [25]. Therefore, the overall complexity for finding all $\Theta(\delta)$ using (1) is $O(\Delta \times |X|^3 \times |\bar{Y}|^3 + |\Sigma| \times |X|^2 \times |\bar{Y}|)$. ■

Example 1: Fig. 1(a) is an example of a stochastic automaton G. The set of states is $X = \{0, 1, 2\}$ with initial state $x_0 = 0$ and event set $\Sigma = \{a, b, c\}$. A state is depicted as a node, whereas a transition is depicted as an edge between its origin and termination states, with its event name and

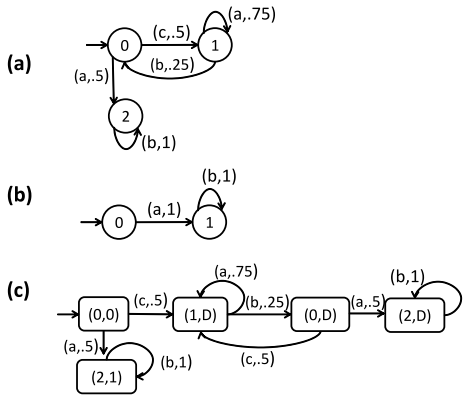


Fig. 1. (a) Stochastic automaton G. (b) Deterministic secret specification R. (c) Refinement G^R.

probability value labeled on the edge. The observation mask M is such that $M(c) = \epsilon$ and for all other events $\sigma \in \{a, b\}$, $M(\sigma) = \sigma$. Suppose R is given in Fig. 1(b), i.e., $K = L(R) = ab^*, L - K = ca^* \cup (ca^*b)^+ \cup (ca^*b)^+ab^*$. Then the refinement G^R automaton is shown in Fig. 1(c). Let the state space of G^R be indexed as the following order: $\{(0, 0), (2, 1), (1, D), (0, D), (2, D)\}$. Then, by solving (1), we get

$$\Theta(a) = \begin{bmatrix} 0 & 0.5 & 0.375 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.75 & 0 & 0 \\ 0 & 0 & 0.375 & 0 & 0.5 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\Theta(b) = \begin{bmatrix} 0 & 0 & 0 & 0.125 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.25 & 0 \\ 0 & 0 & 0 & 0.125 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

■

Remark 2: In this paper, the secret behaviors are specified through a deterministic automaton R, instead of through the subset of the system states. This is because in general, secret may not be captured as a subset of the system model, but a subset of its *behavior*. Moreover, secret can be defined and modeled independently from the plant modeling. In both cases, the secret is specified by R, which is independent of the system model G, while the refined plant G^R with dump state D can be used to capture the violation of secret.

As a simple illustrative example, in Fig. 1, the system state "2" can be either secret or nonsecret, depending on the traces executed before reaching state "2." In this example, the secret specification is defined by R, which models the secret behavior, while the refined plant G^R captures the violation of R, e.g., state "(2, 1)" in G^R denotes the nonviolation of secrecy, while state "(2, D)" denotes violation.

Note that the results developed in this paper can be straightforwardly applied to the case in which secrecy is modeled as a subset of the system states. ■

In [8], a logical version of secrecy was defined, which is satisfied whenever each secret can be masked by a cover and vice versa, with nonzero probability. A weaker version considered in [12] allows some secrets/covers to be nonmasked, but limits the probability of such traces to be a small number. In the next section, a new approach for measuring the level of secrecy is introduced utilizing the notion of JSD, following the review of related information theoretic notions.

C. Information Theoretic Notations

This section reviews related information theoretic notations. As is standard in information theory, a base-2 logarithm has been used. For more details, the reader is referred to [26].

Given a probability distribution p over discrete set A, the entropy of p is defined as

$$H(p) = -\sum_{a \in A} p(a) \log p(a).$$

Given two probability distributions p and q over A, the Kullback–Leibler (KL) divergences between p and q, denoted by $D_{KL}(p, q)$, is defined as

$$D_{KL}(p, q) = \sum_{a \in A} p(a) \log \frac{p(a)}{q(a)}.$$

For given $\lambda_1 > 0$ and $\lambda_2 > 0$ satisfying $\lambda_1 + \lambda_2 = 1$, the JSD between p and q, denoted by $D(p, q)$, is defined as

$$D(p, q) = \lambda_1 D_{KL}(p, \lambda_1 p + \lambda_2 q) + \lambda_2 D_{KL}(q, \lambda_1 p + \lambda_2 q)$$

which is equivalent to

$$D(p, q) = H(\lambda_1 p + \lambda_2 q) - \lambda_1 H(p) - \lambda_2 H(q). \quad (2)$$

Given two probability distributions p over A and q over B, the mutual information between p and q is defined as

$$I(p, q) = \sum_{a \in A, b \in B} \Pr(a, b) \log \frac{\Pr(a, b)}{p(a)q(b)}.$$

Mutual information can also be equivalently defined as

$$I(p, q) = H(p) - H(p|q)$$

where the condition entropy $H(p|q)$ is given as

$$H(p|q) = -\sum_{a \in A} p(a) \sum_{b \in B} \Pr(b|a) \log \Pr(b|a).$$

III. Divergence-Based Secrecy Quantification

For any $n \in \mathbb{N}$ and a length-n observation $o \in \Delta^n$, let $p_n(o)$ denote the probability of observation o. Since the occurrences of observations of length n are mutually disjoint, $\sum_{o \in \Delta^n} p_n(o) = 1$, i.e., p_n is a probability distribution over Δ^n. Then its entropy is given as

$$H(p_n) = -\sum_{o \in \Delta^n} p_n(o) \log p_n(o).$$

Lemma 1: The entropy p_n as defined above for length-n observation can be recursively computed as follows:

$$H(p_n) = H(p_{n-1}) - \sum_{o \in \Delta^{n-1}} p_{n-1}(o) \sum_{\delta \in \Delta} \Pr(\delta|o) \log \Pr(\delta|o).$$

Proof: See the Appendix. ■

Observations in Δ^n can be generated by secrets (behaviors in K) or by covers (behaviors in $L-K$), and accordingly define two more probability distributions over Δ^n: probability that an observation $o \in \Delta^n$ is generated by some secret in K, denoted by $p_n^s(o)$, versus that is generated by some cover in $L-K$, denoted by $p_n^c(o)$

$$p_n^s(o) := \frac{\Pr(s \in K \cap M^{-1}(o))}{\Pr(s \in K \cap M^{-1}(\Delta^n))}$$
$$p_n^c(o) := \frac{\Pr(s \in (L-K) \cap M^{-1}(o))}{\Pr(s \in (L-K) \cap M^{-1}(\Delta^n))}.$$

Further define $\lambda_n^s := \Pr(s \in K \cap M^{-1}(\Delta^n))$ to be the probability of secrets and $\lambda_n^c := \Pr(s \in (L-K) \cap M^{-1}(\Delta^n))$ to be the probability of covers, respectively, generating length-n observation. Then it is easy to show that $\lambda_n^s + \lambda_n^c = 1$ for all $n \in \mathbb{N}$. The entropy of p_n^s and p_n^c are given, respectively, by

$$H\left(p_n^s\right) = -\sum_{o \in \Delta^n} p_n^s(o) \log p_n^s(o) \tag{3}$$
$$H\left(p_n^c\right) = -\sum_{o \in \Delta^n} p_n^c(o) \log p_n^c(o). \tag{4}$$

The ability of an intruder to identify secret versus cover behaviors based on observations of length n depends on the disparity between the two distributions p_n^s versus p_n^c: if p_n^s and p_n^c are identical, i.e., with "zero disparity," there is no way to statistically tell apart secrets from covers, and in that case, there is perfect secrecy. However, when p_n^s and p_n^c are different, then one could characterize the ability of an intruder to discriminate secrets from covers based on length-n observations, using the JSD between p_n^s and p_n^c, denoted by $D(p_n^s, p_n^c)$. This JSD is given by the following weighted sum of a pair of KL divergences between, respectively, p_n^s and p_n^c and their weighted sum:

$$\begin{aligned} D\left(p_n^s, p_n^c\right) &= \lambda_n^s D_{KL}\left(p_n^s, \lambda_n^s p_n^s + \lambda_n^c p_n^c\right) \\ &\quad + \lambda_n^c D_{KL}\left(p_n^c, \lambda_n^s p_n^s + \lambda_n^c p_n^c\right) \\ &= H\left(\lambda_n^s p_n^s + \lambda_n^c p_n^c\right) - \lambda_n^s H\left(p_n^s\right) - \lambda_n^c H\left(p_n^c\right) \\ &= H(p_n) - \lambda_n^s H\left(p_n^s\right) - \lambda_n^c H\left(p_n^c\right) \end{aligned} \tag{5}$$

where D_{KL} represents the KL divergence. Note that JSD is symmetric in its arguments and bounded by 0 and 1.

We first show that the JSD measure as considered in this paper is indeed a useful measure of information revealed, by formally establishing in the following theorem that it equals the mutual information between the observations p_n and the status (secret versus cover) of system executions. (Mutual information is a well-accepted measure of the information revealed about one random variable from the observations of another.) This status can be captured by a bivalued random variable Λ_n, defined for each $n \in N$, such that $\Pr(\Lambda_n = s) = \lambda_n^s$ and $\Pr(\Lambda_n = c) = \lambda_n^c$. With a slight abuse of notation, also denote its distribution by Λ_n, which corresponds to the distribution of the system executing secret versus cover.

Theorem 1: The JSD as defined in (5) is equivalent to the mutual information of Λ_n and p_n, that is

$$D\left(p_n^s, p_n^c\right) = I(\Lambda_n, p_n).$$

Proof: See the Appendix. ■

Remark 3: Theorem 1 establishes the equivalence of the JSD in (5) and the mutual information of Λ_n and p_n, the latter of which measures the mutual dependence between length-n observations and status of system execution (secret versus cover). When $D(p_n^s, p_n^c) = I(\Lambda_n, p_n) = 0$, length-$n$ observations are independent of system execution status, and thus no secret information can be leaked through length-n observations. On the other hand, when $D(p_n^s, p_n^c) = I(\Lambda_n, p_n) > 0$, the dependence of length-n observations and system status can be measured by the JSD, $D(p_n^s, p_n^c)$, which in turn quantifies the extent to which system secrecy can be leaked by length-n observations. ■

A. Recursive Characterization

An intruder is likely to discriminate more if he/she observes for a longer period. Accordingly, our goal is to evaluate the worst case loss of secrecy, as obtained in the limit: $\lim_{n\to\infty} D(p_n^s, p_n^c)$. This worst case JSD provides an upper bound to quantify the amount of information leaked about secrets.

To compute the worst case loss of secrecy, we first develop a recursive computation for $D(p_n^s, p_n^c)$, relating it to distributions of length-$(n-1)$ observations and divergence of length-1 distributions. For $o \in \Delta^*$ and $\delta \in \Delta$, define the distributions of secret versus cover upon a single observation δ following a history of observation o:

$$p^{s|o}(\delta) := \frac{\Pr(s \in K \cap M^{-1}(o\delta))}{\Pr(s \in K \cap M^{-1}(o\{\Delta\}))}$$
$$p^{c|o}(\delta) := \frac{\Pr(s \in (L-K) \cap M^{-1}(o\delta))}{\Pr(s \in (L-K) \cap M^{-1}(o\{\Delta\}))}.$$

Further define $\lambda^{c|o} := \Pr(s \in K \cap M^{-1}(o\{\Delta\}))/\Pr(o)$ and $\lambda^{s|o} := \Pr(s \in (L-K) \cap M^{-1}(o\{\Delta\}))/\Pr(o)$. Then again, we have $\lambda^{c|o} + \lambda^{s|o} = 1$. Following (2), we have:

$$\begin{aligned} D(p^{s|o}, p^{c|o}) &= H(\lambda^{s|o} p^{s|o} + \lambda^{c|o} p^{c|o}) \\ &\quad - \lambda^{s|o} H(p^{s|o}) - \lambda^{c|o} H(p^{c|o}). \end{aligned} \tag{6}$$

The following lemma characterizes the computation of the length-1 JSD for the given observation o.

Lemma 2: Given observation o, the length-1 JSD between $p^{s|o}$ and $p^{c|o}$ can be computed as

$$\begin{aligned} D(p^{s|o}, p^{c|o}) &= H(\lambda^{s|o} p^{s|o} + \lambda^{c|o} p^{c|o}) + H(\{\lambda^{s|o}, \lambda^{c|o}\}) \\ &\quad - H(\lambda^{s|o} p^{s|o}) - H(\lambda^{c|o} p^{c|o}). \end{aligned} \tag{7}$$

Proof: See the Appendix. ■

Using the above lemma, we can next provide the following recursive computation for JSD.

Lemma 3: The JSD over the distribution of length-n secret behaviors and that of cover behaviors can be recursively computed by

$$\begin{aligned} D\left(p_n^s, p_n^c\right) &= H\left(\left\{\lambda_n^s, \lambda_n^c\right\}\right) + \sum_{o \in \Delta^{n-1}} p_{n-1}(o) \\ &\quad \times [-H(\{\lambda^{s|o}, \lambda^{c|o}\}) + D(p^{s|o}, p^{c|o})]. \end{aligned} \tag{8}$$

Proof: See the Appendix. ■

B. State-Distribution-Based Characterization

In order to numerically compute JSD, we map the JSD computation to a computation based on the state-distribution following an observation. Each observation $o \in \Delta^*$ results in a conditional state distribution $\pi(o)$, which can be computed recursively as follows: for any $o \in \Delta^*, \delta \in \Delta$: $\pi(\epsilon) = \pi_0$ and $\pi(o\delta) = (\pi(o) \times \Theta(\delta))/(||\pi(o) \times \Theta(\delta)||)$ [20], where π_0 is the initial state distribution. Let Π denote the set of all such conditional state distributions, and for each $\pi \in \Pi$ and $n \in \mathbb{N}$, denote $P_n(\pi) = \Pr(o \in \Delta^n : \pi(o) = \pi)$, which is the probability that the set of all observations of length n, upon which is the conditional state distribution, is π. Given π, define the following notations:

$$\lambda^{s|\pi} := \sum_{\delta \in \Delta} \pi\,\Theta(\delta)\mathcal{I}^s$$

$$\lambda^{c|\pi} := \sum_{\delta \in \Delta} \pi\,\Theta(\delta)\mathcal{I}^c$$

$$p^{s|\pi}(\delta) := \frac{\pi\,\Theta(\delta)\mathcal{I}^s}{\lambda^{s|\pi}}$$

$$p^{c|\pi}(\delta) := \frac{\pi\,\Theta(\delta)\mathcal{I}^c}{\lambda^{c|\pi}}$$

where $\mathcal{I}^s$ and $\mathcal{I}^c$ denote indicator column vectors of the same size as the number of states, with binary entries to identify the secret versus cover states (states reached by traces in K versus $L - K$). Similar to (6), the length-1 JSD, conditioned upon a current state distribution π, is given by

$$D(p^{s|\pi}, p^{c|\pi}) = H(\lambda^{s|\pi} p^{s|\pi} + \lambda^{c|\pi} p^{c|\pi}) - \lambda^{s|\pi} H(p^{s|\pi}) - \lambda^{c|\pi} H(p^{c|\pi}).$$

Following the definitions introduced above and the recursion result in Lemma 3, the next lemma can be obtained, which characterizes the recursive computation of JSD based on distributions over system state space.

Lemma 4: For a stochastic DES G and specification R, the JSD over the distribution of length-n secret behaviors and that of cover behaviors, as given in (5), can be rewritten as

$$D(p_n^s, p_n^c) = H(\{\lambda_n^s, \lambda_n^c\}) + \sum_{\pi \in \Pi} P_{n-1}(\pi) \times [-H(\{\lambda^{s|\pi}, \lambda^{c|\pi}\}) + D(p^{s|\pi}, p^{c|\pi})]. \quad (9)$$

In the limit when $n \to \infty$, if the distribution $P_n(\cdot)$ over Π converges to $P^*(\cdot)$, then $\lim_{n\to\infty} D(p_n^s, p_n^c)$ exists. The reader is referred to [27] for a condition under which such a convergence is guaranteed: it requires the system to be ergodic (period equals 1 and irreducible) and the existence of a finite sequence $e_1, \ldots, e_m$ such that $\Theta(e_1) \ldots \Theta(e_2)$ is a nonzero subrectangular matrix.

IV. Observer-Based Computation of Worst-Case Secrecy Loss

The state-based characterization of $\lim_{n\to\infty} D(p_n^s, p_n^c)$ requires the computation of $\lim_{n\to\infty} P_{n-1}(\pi)$, which can be accomplished with the help of an observer that was previously introduced in [16] in the context of stochastic diagnosability. An observer tracks the possible system states following each observation and also allows the computation of the corresponding state distribution. We let Obs denote an observer automaton with state set as the power set of states of the refined plant, namely, $Z \subseteq 2^{X \times \bar{Y}}$, so that each node $z \in Z$ of the observer is a subset of system states, i.e., $z \subseteq X \times \bar{Y}$, and we use $|z|$ to denote the number of system states in z. Obs is initialized at node $z_0 = \{(x_0, y_0)\}$ and there is a transition labeled with $\delta \in \Delta$ from node z to z' if and only if every element of z' is reachable from some elements of z along a trace that ends in the only observation δ, i.e., $z' = \{(x', \bar{y}') \in X \times \bar{Y} : \exists (x, \bar{y}) \in z, L_{G^R}((x, \bar{y}), \delta, (x', \bar{y}')) \neq \emptyset\}$. Associated with this transition is the transition probability matrix $\Theta_{z,\delta,z'}$ of size $|z|$ by $|z'|$ (a submatrix of $\Theta(\delta)$ matrix introduced earlier), whose ijth element $\theta_{i,\delta,j}$ is given by the transition probability from the ith element $(x, \bar{y})$ of z to the jth element $(x', \bar{y}')$ of z' while producing the observation δ, and equals $\alpha(L_{G^R}((x, \bar{y}), \delta, (x', \bar{y}')))$.

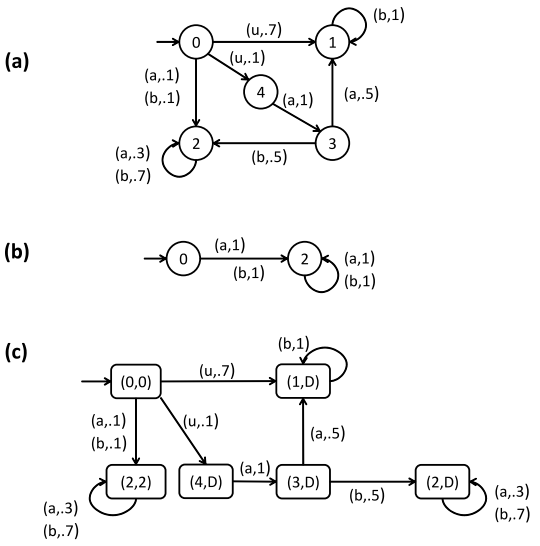


Fig. 2. (a) System model G. (b) Deterministic specification R. (c) Refinement G^R.

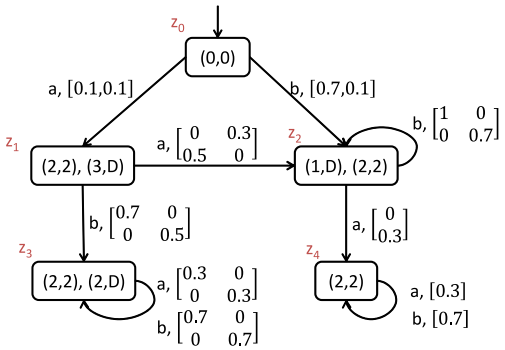


Fig. 3. Observer for the system in Fig. 2.

Example 2: Consider the models of Fig. 2, where $M(u) = \epsilon$, $M(a) = a$, and $M(b) = b$. Then the observer Obs is given as Fig. 3. ■

Associated with each observation $o \in \Delta^*$, there is a reachable state distribution $\pi(o)$ as discussed earlier. Let the

state z be reached in Obs following observation o. Then obviously the number of positive elements of $\pi(o)$ is the same as the number of elements in z. Then with a slight abuse of notation, we also use $\pi(o)$ to denote the row vector containing only positive elements and of the same size as the number of elements in the node reached by o in Obs. Then $\pi(o)$ can also be recursively computed as follows: for any $o \in \Delta^*, \delta \in \Delta$: $\pi(\epsilon) = 1$ and $\pi(o\delta) = (\pi(o) \times \Theta_{z_o,\delta,z_{o\delta}})/(||\pi(o) \times \Theta_{z_o,\delta,z_{o\delta}}||)$, where z_o and $z_{o\delta}$ are the nodes reached in Obs following o and $o\delta$, respectively. Then it can be seen that along any cycle in Obs, the distribution upon completing the cycle is a function of the distribution upon entering the cycle, through a sequence of transition matrix multiplications and their normalization. In the case of steady state, those two distributions will be the same, namely, a fixed point of that function.

Given the Obs with state space Z for system G and specification R, let $\widetilde{\Theta}$ be a $\left(\sum_z |z|\right) \times \left(\sum_z |z|\right)$ square matrix, whose ijth block is the $|z_i| \times |z_j|$ matrix $\sum_\delta \Theta_{z_i,\delta,z_j}$. The fix point distribution associated with $\widetilde{\Theta}$ can be obtained by solving $\pi^* = \pi^*\widetilde{\Theta}$, where π^* is a row vector of size $\sum_z |z|$. For each $z_i \in Z$, let $p(z_i)$ be the summation of the ith block of π^*, then z_i is said to be *recurrent* if $p(z_i) > 0$. Note here that z_i is recurrent if and only if there exists $(x, \bar{y}) \in z_i$ such that $(z, (x, \bar{y}))$ is recurrent in $\widetilde{\Theta}$ as defined in Section II-A. Also note that for each $z \in Z$, there exists a sufficiently large N such that $p(z) = \sum_{o \in \Delta^N:\ o \text{ reaches } z} p_N(o)$. In other words, $p(z)$ is the probability of all sufficiently long observations that reach the observer state z. With a slight abuse of notations, define λ^s as the summation of the elements of π^* corresponding to secret states, i.e., $\lambda^s := \pi^*\mathcal{I}^s$, and $\lambda^c = 1 - \lambda^s$.

Example 3: For the observer in Example 2, $\sum_z |z| = 8$ and so $\widetilde{\Theta}$ is a 8×8 matrix given as

$$\widetilde{\Theta} = \begin{bmatrix} 0 & 0.1 & 0.1 & 0.7 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.3 & 0.7 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.7 & 0 & 0 & 0.3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and

$$\pi^* = [0 \quad 0 \quad 0 \quad 0.75 \quad 0 \quad 0.07 \quad 0.05 \quad 0.13].$$

Therefore, $p(z_0) = p(z_1) = 0$, $p(z_2) = 0.75$, $p(z_3) = 0.12$ and $p(z_4) = 0.13$. Since

$$\mathcal{I}^s = [1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1]^T$$
$$\mathcal{I}^c = [0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0]^T$$

we have $\lambda^s = 0.2$ and $\lambda^c = 0.8$. ■

For a set of recurrent nodes $\{z_1, z_2, \ldots, z_n\}$ that forms an SCC in Obs, define a set of distributions $\{\pi^*_{z_1}, \pi^*_{z_2}, \ldots, \pi^*_{z_n}\}$ to be a set of steady-state distributions if $\forall i, j, \delta$ such that Θ_{z_i,δ,z_j} is defined, the following holds:

$$\pi^*_{z_j} = \frac{\pi^*_{z_i}\Theta_{z_i,\delta,z_j}}{||\pi^*_{z_i}\Theta_{z_i,\delta,z_j}||}$$

i.e., $\pi^*_{z_i}$ represents a steady-state conditional distribution following a single sufficiently long observation o that reaches z_i. Note that in this case, any other extension of o that also reaches z_i will induce the same conditional distribution $\pi^*_{z_i}$. There may exist multiple sets of steady-state distributions for a given set of recurrent nodes, denoted by $\{\{\pi^*_{z_1,k}, \ldots, \pi^*_{z_n,k}\}, k \in \mathbb{N}\}$. Then if steady state always exists, for any sufficiently long observation that reaches a recurrent node z, there exists $k \in \mathbb{N}$ such that $\pi(o) = \pi^*_{z,k}$. Denote $p(z,k) := \Pr[\{o \mid o \text{ reaches } z \text{ and } \pi(o) = \pi^*_{z,k}\}]$. Note that when the set of steady state distributions is a singleton and hence unique, $p(z,k) = p(z)$.

Example 4: Let us revisit Example 2. It can be seen that z_2, z_3, and z_4 are recurrent nodes, and each of them forms an SCC. We have $\pi^*_{z_2} = [1\ 0]$ and $\pi^*_{z_4} = [1]$, and while there are multiple solutions to the equation set $\pi^*_{z_3} = (\pi^*_{z_3}\Theta_{z_3,a,z_3})/(\pi^*_{z_3}\Theta_{z_3,a,z_3})$ and $\pi^*_{z_3} = (\pi^*_{z_3}\Theta_{z_3,b,z_3})/(\pi^*_{z_3}\Theta_{z_3,b,z_3})$, only $\pi^*_{z_3} = [0.5833\ 0.4167]$ is reachable. Thus, each set of recurrent nodes is a singleton set, each with a unique fixed-point distribution. Therefore, for each recurrent node z, $p(z,k) = p(z)$. ■

Let $\mathcal{I}^s_{z'}$ and $\mathcal{I}^c_{z'}$ be indicator column vectors with binary entries of size $|z'|$ for identifying, within z', the secret and cover states, respectively. For each steady-state distribution $\pi^*_{z,k}$ of each recurrent node z, define

$$\lambda^{s|\pi^*_{z,k}} := \sum_{\delta\in\Delta} \pi^*_{z,k}\Theta_{z,\delta,z'}\mathcal{I}^s_{z'}$$
$$\lambda^{c|\pi^*_{z,k}} := \sum_{\delta\in\Delta} \pi^*_{z,k}\Theta_{z,\delta,z'}\mathcal{I}^c_{z'}$$
$$p^{s|\pi^*_{z,k}}(\delta) := \frac{\pi^*_{z,k}\Theta_{z,\delta,z'}\mathcal{I}^s_{z'}}{\lambda^{s|\pi^*_{z,k}}}$$
$$p^{c|\pi^*_{z,k}}(\delta) := \frac{\pi^*_{z,k}\Theta_{z,\delta,z'}\mathcal{I}^c_{z'}}{\lambda^{c|\pi^*_{z,k}}}.$$

Example 5: Then for Example 2, $\mathcal{I}^s_{z_2} = [0\ 1]^T$, $\mathcal{I}^c_{z_2} = [1\ 0]^T$, $\mathcal{I}^s_{z_3} = [1\ 0]^T$, $\mathcal{I}^c_{z_3} = [0\ 1]^T$, $\mathcal{I}^s_{z_4} = [1]^T$, and $\mathcal{I}^c_{z_4} = [0]^T$. For z_2 and $\pi^*_{z_2}$

$$\lambda^{s|\pi^*_{z_2}} = 0$$
$$\lambda^{c|\pi^*_{z_2}} = 1$$
$$p^{c|\pi^*_{z_2}}(b) = \frac{\pi^*_{z_2}\Theta_{z_2,b,z_2}\mathcal{I}^c_{z_2}}{\lambda^{c|\pi^*_{z_2}}} = 1$$
$$p^{s|\pi^*_{z_2}}(a) = p^{c|\pi^*_{z_2}}(a) = p^{s|\pi^*_{z_2}}(b) = 0.$$

For z_3 and $\pi^*_{z_3}$

$$\lambda^{s|\pi^*_{z_3}} = 0.5833$$
$$\lambda^{c|\pi^*_{z_3}} = 0.4167$$
$$p^{s|\pi^*_{z_3}}(a) = \frac{\pi^*_{z_3}\Theta_{z_3,a,z_3}\mathcal{I}^s_{z_3}}{\lambda^{s|\pi^*_{z_3}}} = 0.3$$
$$p^{s|\pi^*_{z_3}}(b) = \frac{\pi^*_{z_3}\Theta_{z_3,b,z_3}\mathcal{I}^s_{z_3}}{\lambda^{s|\pi^*_{z_3}}} = 0.7$$
$$p^{c|\pi^*_{z_3}}(a) = \frac{\pi^*_{z_3}\Theta_{z_3,a,z_3}\mathcal{I}^c_{z_3}}{\lambda^{c|\pi^*_{z_3}}} = 0.3$$
$$p^{c|\pi^*_{z_3}}(b) = \frac{\pi^*_{z_3}\Theta_{z_3,b,z_3}\mathcal{I}^c_{z_3}}{\lambda^{c|\pi^*_{z_3}}} = 0.7.$$

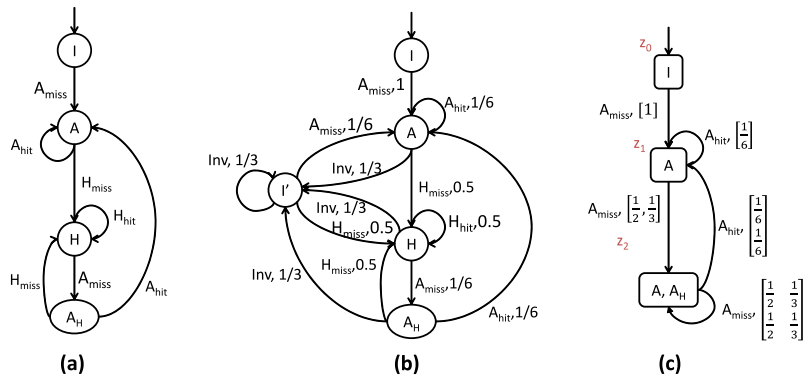


Fig. 4. (a) Cache side-channel attack. (b) Cache side-channel attack with random eviction. (c) Observer for the cache side-channel attack example with random eviction.

For z_4 and π^*_{z4}

$$\lambda^{s|\pi^*_{z4}} = 1$$
$$\lambda^{c|\pi^*_{z4}} = 0$$
$$p^{s|\pi^*_{z4}}(a) = \frac{\pi^*_{z4}\Theta_{z4,a,z4}\mathcal{I}^s_{z4}}{\lambda^{s|\pi^*_{z4}}} = 0.3$$
$$p^{s|\pi^*_{z4}}(b) = \frac{\pi^*_{z4}\Theta_{z4,b,z4}\mathcal{I}^s_{z4}}{\lambda^{s|\pi^*_{z4}}} = 0.7$$
$$p^{c|\pi^*_{z4}}(a) = p^{c|\pi^*_{z4}}(b) = 0.$$

■

In the following, we assume the existence of steady state.

Assumption 1: Assume that for any sufficiently long observations $o_1 \leq o_2$, if Obs reaches the same node following o_1 and o_2, then $\pi(o_1) = \pi(o_2)$.

Then following the above definitions and Lemma 4, the next theorem provides computation of $\lim_{n\to\infty} D(p^s_n, p^c_n)$, under Assumption 1.

Theorem 2: Consider a system G and specification R. Then under Assumption 1, the worst case secrecy loss, i.e., JSD between p^s_n and p^c_n when $n \to \infty$, is given by

$$\lim_{n\to\infty} D(p^s_n, p_n = c) = H(\{\lambda^s, \lambda^c\}) + \sum_{z:z \text{ is recurrent}} \sum_{k\in\mathbb{N}} \times p(z,k)[-H(\{\lambda^{s|\pi^*_{z,k}}, \lambda^{c|\pi^*_{z,k}}\}) + D(p^{s|\pi^*_{z,k}}, p^{c|\pi^*_{z,k}})].$$

The next assumption assumes that for each set of recurrent nodes in Obs, there only exists one set of steady-state distributions.

Assumption 2: For each set of recurrent nodes in Obs, $k = 1$, i.e., the set of steady-state distributions is unique, so that $p(z,k) = p(z)$.

Theorem 3: Consider a system G and specification R. Then under Assumptions 1 and 2, the worst case secrecy loss, i.e., the JSD between p^s_n and p^c_n when $n \to \infty$, is given by

$$\lim_{n\to\infty} D\big(p^s_n, p^c_n\big) = H(\{\lambda^s, \lambda^c\}) + \sum_{z:z \text{ is recurrent}} \times p(z)\big[-H(\{\lambda^{s|\pi^*_z}, \lambda^{c|\pi^*_z}\}) + D(p^{s|\pi^*_z}, p^{c|\pi^*_z})\big].$$

Note that computing limiting JSD requires construction of an observer, whose worst case complexity is exponential to the number of states in G^R [28].

Example 6: Revisit Example 2. Following Examples 2–5, we have:

$$H(\{\lambda^s, \lambda^c\}) = -0.2\log(0.2) - 0.8\log(0.8) = 0.7219$$
$$\times \sum_{z:z \text{ is recurrent}} p(z)\big[-H(\{\lambda^{s|\pi^*_z}, \lambda^{c|\pi^*_z}\}) + D(p^{s|\pi^*_z}, p^{c|\pi^*_z})\big]$$
$$= 0.1176.$$

Therefore, according to Theorem 3

$$\lim_{n\to\infty} D\big(p^s_n, p^c_n\big) = -0.7219 - 0.1176 = 0.6043.$$

Thus, for the system in Fig. 2, the worst case secrecy loss, as measured by the limiting JSD, is 0.6043. ■

Remark 4: As discussed earlier (see Remark 3), the worst case secrecy loss is bounded between 0 and 1, where 0 indicates no secrecy loss and 1 means largest secrecy leaks. As a relative value, the smaller it is, the better system design is regarding protecting system secret.

Note also that current quantification measures how much secrecy is lost in general with respect to the set of secret behaviors specified by automaton R. In case quantification for leaking of each secret behavior is interested, one can construct automaton R_s for each secret behavior s and compute the limiting JSD for each pair of G and R_s. ■

V. Cache Side-Channel Attack Example

In this section, a modified version of cache side-channel attack example adopted from [29] is considered. When a host program executes on the system, its memory accesses contain information that might help an attacker determine the secret of whether or not the host is accessing the cache memory. Suppose the attacker executes a program on the same processor and shares the same cache as the host program [see Fig. 4(a)]. If the host holds its own data in cache, its cache access results in a hit (H_{hit}), but if the attacker evicts the host's data in the cache lines by requesting cache access, it would result a miss (H_{miss}). Similarly, when the host requires cache data,

it evicts the cache lines that hold the attacker's data, which make the attacker's future cache access "miss" (A_{miss}). On the other hand, A_{hit} occurs when the attacker accesses the data while it is held in cache. These behaviors may give the attacker information to infer the host's cache accesses. The system is described as in Fig. 4. Note that as long as the host requests cache access, the attacker will for sure witness an A_{miss} and at that point it would occupy the cache, thereby fully knowing the cache status.

Now suppose, in order to prevent any information leak, the system (i.e., the processor) practices attack protection by periodically evicting the occupant of cache through the execution of the "Inv" event as shown in Fig. 4(b). This introduces ambiguity in the attacker's knowledge about the occupancy of the cache, i.e., when it observes a cache miss, it does not know whether it is due to the processor's eviction or due to the host's cache access. Note that in Fig. 4(a) and (b), the state A_H is used to denote that currently the cache line is filled with attacker's data, while in the last step, it was filled with host's data. Then in Fig. 4(b), we can view $\{H, A_H\}$ to be the secret states, whereas $\{I, A, I'\}$ to be the cover states. Assuming that only the A_{miss} and A_{hit} are observable to the attacker, its observer model Obs is as given in Fig. 4(c). It can be computed that $p(z_1) = 1/6$, $p(z_2) = 5/6$, $\pi^*_{z_1} = [1]$, $\pi^*_{z_2} = [0.6\ 0.4]$, $\lambda_s = 1/3$, and $\lambda_c = 2/3$. Further, the limiting divergence $\lim_{n\to\infty} D(p^s_n, p^c_n) = 0$, meaning that no amount of secrecy could be leaked through the side channel if the cache line is periodically evicted by the processor.

VI. Conclusion

In this paper, we presented an information theoretic measure of secrecy loss in SPODESs, where the information about system secrets may be revealed through the side channel of observable inputs/outputs. Statistical difference, in the form of the JSD measure between the influence of secrets versus covers on the observations, is employed to quantify the loss of secrecy. We showed that this JSD measure is equivalent to the mutual information between the distribution over possible observations and that over possible status of system execution (secret versus cover), and proposed the computation of the "limiting" JSD as a measure of worst case secrecy loss, resulting from their longer and longer observations. The computation of limiting JSD required developing a recursion relating JSD over length-n sequences to distributions over length-(n-1) sequences through the one-step dynamics of underlying system model. We also presented an observer-based approach for computing the fixed-point of recursion and also the limiting JSD. Illustrative examples, including the one based on side-channel attack, are provided to demonstrate the proposed notions and associated computation. For terminating DESs, one can simply add unobservable self-loops at terminating states with probability 1 and proceed as in this paper. Future work can consider generalizing the results, by computing the JSD for systems that may not satisfy the two assumptions about convergence (see Assumption 1) and uniqueness (see Assumption 2), respectively, and extend the results to stochastic hybrid systems [30]. Quantification of secrecy loss that also considers the modeling uncertainties is another future research direction.

Appendix

Proof of Lemma 1: According to the definition of entropy, we have

$$\begin{aligned}
&H(p_n)\\
&= -\sum_{o\in\Delta^n} p_n(o)\log p_n(o)\\
&= -\sum_{o\in\Delta^{n-1}}\sum_{\delta\in\Delta} p_n(o\delta)\log p_n(o\delta)\\
&= -\sum_{o\in\Delta^{n-1}}\sum_{\delta\in\Delta} p_{n-1}(o)\Pr(\delta|o)\log(p_{n-1}(o)\Pr(\delta|o))\\
&= -\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\sum_{\delta\in\Delta}\Pr(\delta|o)(\log p_{n-1}(o)+\log\Pr(\delta|o))\\
&= -\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\sum_{\delta\in\Delta}\Pr(\delta|o)\log\Pr(\delta|o)\\
&\quad -\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\log p_{n-1}(o)\sum_{\delta\in\Delta}\Pr(\delta|o)\\
&= -\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\sum_{\delta\in\Delta}\Pr(\delta|o)\log\Pr(\delta|o)\\
&\quad -\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\log p_{n-1}(o)\\
&= -\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\sum_{\delta\in\Delta}\Pr(\delta|o)\log\Pr(\delta|o)+H(p_{n-1}).
\end{aligned}$$

Thus, Lemma 1 is established. ∎

Proof of Theorem 1: According to the definition of mutual information, we have

$$I(\Lambda_n, p_n) = H(p_n) - H(p_n|\Lambda_n).$$

The conditional entropy $H(p_n|\Lambda_n)$ can be expressed as

$$\begin{aligned}
H(p_n|\Lambda_n) &= -\lambda^s_n\sum_{o\in\Delta^n}\Pr(o|\Lambda_n=s)\log\Pr(o|\Lambda_n=s)\\
&\quad -\lambda^c_n\sum_{o\in\Delta^n}\Pr(o|\Lambda_n=c)\log\Pr(o|\Lambda_n=c)\\
&= -\lambda^s_n\sum_{o\in\Delta^n} p^s_n(o)\log p^s_n(o)\\
&\quad -\lambda^c_n\sum_{o\in\Delta^n} p^c_n(o)\log p^s_n(o)\\
&= \lambda^s_n H\left(p^s_n\right)+\lambda^c_n H\left(p^c_n\right)
\end{aligned}$$

where we utilize the fact that $\Pr(o|\Lambda_n = s) = p^s_n(o)$ and $\Pr(o|\Lambda_n = c) = p^c_n(o)$. Substituting $H(p_n|\Lambda_n)$ into the definition of mutual information $I(\Lambda_n, p_n)$ and considering the relationship in (5), we have

$$I(\Lambda_n, p_n) = H(p_n) - \lambda^s_n H\left(p^s_n\right) - \lambda^c_n H\left(p^c_n\right) = D\left(p^s_n, p^c_n\right).$$

Thus, the proof is completed. ∎

Proof of Lemma 2: By expanding (6), we have

$$\begin{aligned}
D(p^{s|o}, p^{c|o}) &= H(\lambda^{s|o}p^{s|o}+\lambda^{c|o}p^{c|o})\\
&\quad +\sum_{\delta\in\Delta}\lambda^{s|o}p^{s|o}(\delta)\log\frac{\lambda^{s|o}p^{s|o}(\delta)}{\lambda^{s|o}}\\
&\quad +\sum_{\delta\in\Delta}\lambda^{c|o}p^{c|o}(\delta)\log\frac{\lambda^{c|o}p^{c|o}(\delta)}{\lambda^{c|o}}
\end{aligned}$$

$$
\begin{aligned}
&= H(\lambda^{s|o}p^{s|o} + \lambda^{c|o}p^{c|o}) \\
&\quad + \sum_{\delta\in\Delta} \lambda^{s|o}p^{s|o}(\delta)\log\lambda^{s|o}p^{s|o}(\delta) \\
&\quad + \sum_{\delta\in\Delta} \lambda^{c|o}p^{c|o}(\delta)\log\lambda^{c|o}p^{c|o}(\delta) \\
&\quad - \lambda^{s|o}\log\lambda^{s|o}\left(\sum_{\delta\in\Delta} p^{s|o}(\delta)\right) \\
&\quad - \lambda^{c|o}\log\lambda^{c|o}\left(\sum_{\delta\in\Delta} p^{c|o}(\delta)\right).
\end{aligned}
$$

Since $\left(\sum_{\delta\in\Delta} p^{s|o}(\delta)\right) = \left(\sum_{\delta\in\Delta} p^{c|o}(\delta)\right) = 1$, we have

$$
\begin{aligned}
D(p^{s|o}, p^{c|o}) &= H(\lambda^{s|o}p^{s|o} + \lambda^{c|o}p^{c|o}) \\
&\quad + \sum_{\delta\in\Delta} \lambda^{s|o}p^{s|o}(\delta)\log\lambda^{s|o}p^{s|o}(\delta) \\
&\quad + \sum_{\delta\in\Delta} \lambda^{c|o}p^{c|o}(\delta)\log\lambda^{c|o}p^{c|o}(\delta) \\
&\quad - \lambda^{s|o}\log\lambda^{s|o} - \lambda^{c|o}\log\lambda^{c|o} \\
&= H(\lambda^{s|o}p^{s|o} + \lambda^{c|o}p^{c|o}) + H(\{\lambda^{s|o}, \lambda^{c|o}\}) \\
&\quad - H(\lambda^{s|o}p^{s|o}) - H(\lambda^{c|o}p^{c|o}).
\end{aligned}
$$

Thus, Lemma 2 is established. ∎

Proof of Lemma 3: We first define some notations, for simplicity of presentation, as follows:

$$
\begin{aligned}
\widetilde{p}^s_n(o) &:= \Pr(s \in K \cap M^{-1}(o)) = \lambda^s_n p^s_n(o) \\
\widetilde{p}^c_n(o) &:= \Pr(s \in (L-K) \cap M^{-1}(o)) = \lambda^c_n p^c_n(o) \\
\widetilde{p}^s(\delta|o) &:= \frac{\Pr(s \in K \cap M^{-1}(o\delta))}{\Pr(o)} = \lambda^{s|o}p^{s|o}(\delta) \\
\widetilde{p}^c(\delta|o) &:= \frac{\Pr(s \in (L-K) \cap M^{-1}(o\delta))}{\Pr(o)} = \lambda^{c|o}p^{c|o}(\delta) \\
p(\delta|o) &:= \lambda^{s|o}p^{s|o}(\delta) + \lambda^{c|o}p^{c|o}(\delta) = \Pr(\delta|o).
\end{aligned}
$$

We start by deriving a recursive computation for $H(p^s_n)$ as defined in (3) as follows:

$$
\begin{aligned}
H\left(p^s_n\right) &= -\sum_{o\in\Delta^n} p^s_n(o)\log p^s_n(o) \\
&= -\frac{1}{\lambda^s_n}\sum_{o\in\Delta^n} \widetilde{p}^s_n(o)\log\frac{\widetilde{p}^s_n(o)}{\lambda^s_n} \\
&= -\frac{1}{\lambda^s_n}\sum_{o\in\Delta^{n-1}}\sum_{\delta\in\Delta} \widetilde{p}^s_n(o\delta)\log\frac{\widetilde{p}^s_n(o\delta)}{\lambda^s_n} \\
&= -\frac{1}{\lambda^s_n}\sum_{o\in\Delta^{n-1}}\sum_{\delta\in\Delta} p_{n-1}(o)\widetilde{p}^s(\delta|o)\log\frac{p_{n-1}(o)\widetilde{p}^s(\delta|o)}{\lambda^s_n} \\
&= -\frac{1}{\lambda^s_n}\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\sum_{\delta\in\Delta}\widetilde{p}^s(\delta|o) \\
&\quad \times\left[\log p_{n-1}(o) + \log(p^{s|o}(\delta)\lambda^{s|o}) - \log\lambda^s_n\right] \\
&= -\frac{1}{\lambda^s_n}\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\log p_{n-1}(o)\sum_{\delta\in\Delta}\widetilde{p}^s(\delta|o) \\
&\quad -\frac{1}{\lambda^s_n}\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\sum_{\delta\in\Delta}\widetilde{p}^s(\delta|o)\log(p^{s|o}(\delta)\lambda^{s|o}) \\
&\quad +\frac{1}{\lambda^s_n}\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\sum_{\delta\in\Delta}\widetilde{p}^s(\delta|o)\log\lambda^s_n \\
&= -\frac{1}{\lambda^s_n}\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\log p_{n-1}(o)\lambda^{s|o} + \log\lambda^s_n \\
&\quad +\frac{1}{\lambda^s_n}\sum_{o\in\Delta^{n-1}} p_{n-1}(o)H(\lambda^{s|o}p^{s|o}).
\end{aligned}
$$

Similarly, $H(p^c_n)$ as defined in (4) can be recursively characterized as

$$
\begin{aligned}
H\left(p^c_n\right) &= -\frac{1}{\lambda^c_n}\sum_{o\in\Delta^{n-1}} p_{n-1}(o)\log p_{n-1}(o)\lambda^{c|o} + \log\lambda^c_n \\
&\quad +\frac{1}{\lambda^c_n}\sum_{o\in\Delta^{n-1}} p_{n-1}(o)H(\lambda^{c|o}p^{c|o}).
\end{aligned}
$$

Expanding (5) using the above recursion and Lemma 1 yields the following:

$$
\begin{aligned}
&D\left(p^s_n, p^c_n\right) \\
&= H(p_n) - \lambda^s_n H\left(p^s_n\right) - \lambda^c_n H\left(p^c_n\right) \\
&= H(p_n) - \lambda^s_n\log\lambda^s_n - \lambda^c_n\log\lambda^c_n \\
&\quad + \sum_{o\in\Delta^{n-1}} p_{n-1}(o)\log p_{n-1}(o)\lambda^{s|o} \\
&\quad - \sum_{o\in\Delta^{n-1}} p_{n-1}(o)H(\lambda^{s|o}p^{s|o}) \\
&\quad + \sum_{o\in\Delta^{n-1}} p_{n-1}(o)\log p_{n-1}(o)\lambda^{c|o} \\
&\quad - \sum_{o\in\Delta^{n-1}} p_{n-1}(o)H(\lambda^{c|o}p^{c|o}) \\
&= H\left(\left\{\lambda^s_n, \lambda^c_n\right\}\right) + \sum_{o\in\Delta^{n-1}} p_{n-1}(o) \\
&\quad \times\left[-\sum_{\delta\in\Delta} p(\delta|o)\log p(\delta|o) - H(\lambda^{s|o}p^{s|o}) - H(\lambda^{c|o}p^{c|o})\right] \\
&= H\left(\left\{\lambda^s_n, \lambda^c_n\right\}\right) + \sum_{o\in\Delta^{n-1}} p_{n-1}(o) \\
&\quad \times[H(\lambda^{s|s}p^{s|s} + \lambda^{c|o}p^{c|o}) - H(\lambda^{s|o}p^{s|o}) - H(\lambda^{c|o}p^{c|o})].
\end{aligned}
$$

Finally, by substituting (7) in Lemma 2, we have

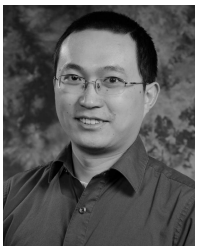
$$
\begin{aligned}
D\left(p^s_n, p^c_n\right) &= H\left(\left\{\lambda^s_n, \lambda^c_n\right\}\right) + \sum_{o\in\Delta^{n-1}} p_{n-1}(o) \\
&\quad \times[-H(\{\lambda^{s|o}, \lambda^{c|o}\}) + D(p^{s|o}, p^{c|o})].
\end{aligned}
$$

Thus, Lemma 3 is established. ∎

REFERENCES

[1] D. Kundur and K. Ahsan, "Practical Internet steganography: Data hiding in IP," in *Proc. Texas Workshop Secur. Inf. Syst.*, College Station, TX, USA, Apr. 2003, pp. 1–5.

[2] C. S. Collberg and C. Thomborson, "Watermarking, tamper-proofing, and obfuscation—Tools for software protection," *IEEE Trans. Softw. Eng.*, vol. 28, no. 8, pp. 735–746, Aug. 2002.
[3] J. Ren and J. Wu, "Survey on anonymous communications in computer networks," *Comput. Commun.*, vol. 33, no. 4, pp. 420–431, Mar. 2010.
[4] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters, "Candidate indistinguishability obfuscation and functional encryption for all circuits," in *Proc. 54th Annu. IEEE Symp. Found. Comput. Sci. (FOCS)*, Berkeley, CA, USA, Oct. 2013, pp. 40–49.
[5] G. Smith, "On the foundations of quantitative information flow," in *Proc. Int. Conf. Found. Softw. Sci. Comput. Struct.*, 2009, pp. 288–302.
[6] M. Backes, B. Köpf, and A. Rybalchenko, "Automatic discovery and quantification of information leaks," in *Proc. 30th IEEE Symp. Secur. Privacy*, Washington, DC, USA, May 2009, pp. 141–153.
[7] B. Espinoza and G. Smith, "Min-entropy as a resource," *Inf. Comput.*, vol. 226, pp. 57–75, May 2013.
[8] S. Takai and R. Kumar, "Verification and synthesis for secrecy in discrete-event systems," in *Proc. Amer. Control Conf.*, St. Louis, MO, USA, Jun. 2009, pp. 4741–4746.
[9] J. Bryans, M. Koutny, and C. Mu, "Towards quantitative analysis of opacity," Dept. Comput. Sci., Newcastle Univ., Newcastle upon Tyne, U.K., Tech. Rep. CS-TR-1304, Nov. 2011.
[10] A. Saboori and C. N. Hadjicostis, "Probabilistic current-state opacity is undecidable," in *Proc. 19th Int. Symp. Math. Theory Netw. Syst.*, Budapest, Hungary, Jul. 2010, pp. 477–483.
[11] A. Saboori and C. N. Hadjicostis, "Current-state opacity formulations in probabilistic finite automata," *IEEE Trans. Autom. Control*, vol. 59, no. 1, pp. 120–133, Jan. 2014.
[12] M. Ibrahim, J. Chen, and R. Kumar, "Secrecy in stochastic discrete event systems," in *Proc. 11th IEEE Int. Conf. Netw., Sens. Control*, Miami, FL, USA, Apr. 2014, pp. 48–53.
[13] A. Saboori and C. N. Hadjicostis, "Verification of K-step opacity and analysis of its complexity," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 3, pp. 549–559, Jul. 2011.
[14] A. Saboori and C. N. Hadjicostis, "Verification of infinite-step opacity and complexity considerations," *IEEE Trans. Autom. Control*, vol. 57, no. 5, pp. 1265–1269, May 2012.
[15] R. Jacob, J.-J. Lesage, and J.-M. Faure, "Opacity of discrete event systems: Models, validation and quantification," in *Proc. 5th Int. Workshop Dependable Control Discrete Syst.*, Cancún, Mexico, May 2015, pp. 174–181.
[16] D. Thorsley and D. Teneketzis, "Diagnosability of stochastic discrete-event systems," *IEEE Trans. Autom. Control*, vol. 50, no. 4, pp. 476–492, Apr. 2005.
[17] B. Bérard, K. Chatterjee, and N. Sznajder, "Probabilistic opacity for Markov decision processes," *Inf. Process. Lett.*, vol. 115, no. 1, pp. 52–59, Jan. 2015.
[18] V. K. Garg, R. Kumar, and S. I. Marcus, "A probabilistic language formalism for stochastic discrete-event systems," *IEEE Trans. Autom. Control*, vol. 44, no. 2, pp. 280–293, Feb. 1999.
[19] A. Xie and P. A. Beerel, "Efficient state classification of finite-state Markov chains," *IEEE Trans. Comput.-Aided Des. Integr.*, vol. 17, no. 12, pp. 1334–1339, Dec. 1998.
[20] J. Chen and R. Kumar, "Failure detection framework for stochastic discrete event systems with guaranteed error bounds," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1542–1553, Jun. 2015.
[21] J. Chen and R. Kumar, "Stochastic failure prognosability of discrete event systems," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1570–1581, Jun. 2015.
[22] J. G. Kemény and J. L. Snell, *Finite Markov Chains*, vol. 356. Princeton, NJ, USA: Van Nostrand, 1960.
[23] X. Wang and A. Ray, "A language measure for performance evaluation of discrete-event supervisory control systems," *Appl. Math. Model.*, vol. 28, no. 9, pp. 817–833, Sep. 2004.
[24] D. Thorsley, T.-S. Yoo, and H. E. Garcia, "Diagnosability of stochastic discrete-event systems under unreliable observations," in *Proc. Amer. Control Conf.*, Seattle, WA, USA, Jun. 2008, pp. 1158–1165.
[25] J. Chen and R. Kumar, "Polynomial test for stochastic diagnosability of discrete-event systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 4, pp. 969–979, Oct. 2013.
[26] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley, 2012.
[27] T. Kaijser, "A limit theorem for partially observed Markov chains," *Ann. Probab.*, vol. 3, no. 4, pp. 677–696, Aug. 1975.
[28] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Berlin, Germany: Springer, 2009.
[29] T. Zhang and R. B. Lee, "Secure cache modeling for measuring side-channel leakage," Dept. Electr. Eng., Princeton Univ., Princeton, NJ, USA, Tech. Rep. 428, 2014.
[30] J. Chen and R. Kumar, "Fault detection of discrete-time stochastic systems subject to temporal logic correctness requirements," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 4, pp. 1369–1379, Oct. 2015.



Jun Chen (S'11–M'14) received the B.S. degree in automation from Zhejiang Uniersity, Hangzhou, China, in 2009, and the Ph.D. degree in electrical engineering from Iowa State University, Ames, IA, USA, in 2014.

He was a Student Intern with General Motors Research and Development, Warre, MI, USA, in 2014. He joined the Idaho National Laboratory, Idaho Falls, ID, USA, in 2014, where he is currently a Research and Development Scientist. His current research interests include discrete-event systems, stochastic hybrid systems, power and energy systems, together with their control, optimization, diagnosis, and resiliency analysis.

Dr. Chen was a recipient of the Best Paper Award from the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, the Publication Achievement Award and the Exceptional Contributions Program Award from the Idaho National Laboratory, the Research Excellence Award from Iowa State University, and the Provost Graduate Fellowship from the University of Central Florida. Since 2013, he has been a TPC Member of the Chinese Control and Decision Conference.

Mariam Ibrahim (M'16) received the B.S. degree in electrical and computer engineering from Hashemite University, Zarqa, Jordan, in 2008, the M.S. degree in mechatronics engineering from Al-Balqa' Applied University, Amman, Jordan, in 2011, and the Ph.D. degree in electrical engineering from Iowa State University, Ames, IA, USA, in 2016, with a scholarship from German Jordanian University, Amman, Jordan.

From 2008 to 2011, she was a Lab Supervisor with the Department of Electrical Engineering, Hashemite University. In 2011, she joined from German Jordanian University as a Research Assistant, where she is currently an Assistant Professor. Her current research interests include discrete-event systems, stochastic systems, power systems, together with their control and resiliency analysis, and model-based verification with AADL.

Ratnesh Kumar (S'87–M'90–SM'00–F'07) received the B.Tech. degree in electrical engineering from IIT Kanpur, Kanpur, India, in 1987, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Texas at Austin, Austin, TX, USA, in 1989 and 1991, respectively.

He was with the Electrical and Computer Engineering Department, University of Kentucky, Lexington, KY, USA, from 1991 to 2002. He has held visiting positions with the University of Maryland, College Park, MD, USA, the Applied Research Laboratory, Penn State University, State College, PA, USA, NASA Ames, Mountain View, CA, USA, the Idaho National Laboratory, Idaho Falls, ID, USA, the United Technologies Research Center, East Hartford, CT, USA, and the Air Force Research Laboratory, Dayton, OH, USA. He has been a Professor with the Electrical and Computer Engineering Department, Iowa State University, Ames, IA, USA, since 2002. His current research interests include sensors, networks, controls, and software with the application domains of safety and security in cyber-physical (hybrid) systems, embedded and real-time systems, model-based software and web-services, data analytics, power grid and energy harvesting, and sustainable agriculture.

Prof. Kumar was a recipient of the Gold Medals for the Best EE Undergraduate and the Best All Rounder from IIT Kanpur, the Best Dissertation Award from the University of Texas at Austin, and the Best Paper Award from the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING. He is a Distinguished Lecturer of the IEEE Control Systems Society and an Associate Editor of the *ACM Transactions on Embedded Computing Systems*, the *SIAM Journal on Control and Optimization*, the IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, *IET Cyber-Physical Systems: Theory & Applications*, the *Journal of Discrete Event Dynamical Systems*, the IEEE Control Systems Society, the IEEE Robotics and Automation Systems Society, and the IEEE Workshop on Software Cybernetics.