Quantification of Centralized/Distributed Secrecy in Stochastic Discrete Event Systems

Mariam Ibrahim, Jun Chen and Ratnesh Kumar

Abstract Unlike information, behaviors cannot be encrypted and may instead be protected by providing covers that generate indistinguishable observations from behaviors needed to be kept secret. Such a scheme may still leak information about secrets due to statistical difference between the occurrence probabilities of the secrets and their covers. Jensen-Shannon Divergence (JSD) is a possible means of quantifying statistical difference between two distributions and can be used to measure such information leak as is presented in this chapter. Using JSD, we quantify loss of secrecy in stochastic partially-observed discrete event systems in two settings: (i) the centralized setting, corresponding to a single attacker/observer, and (ii) the distributed collusive setting, corresponding to multiple attackers/observers, exchanging their observed information. In the centralized case, an observer structure is formed and used to aide the computation of JSD, in the limit, as the length of observations approach infinity to quantify the worst case loss of secrecy. In the distributed collusive case, channel models are introduced to extend the system model to capture the effect of exchange of observations, that allows the JSD computation of the centralized case to be applied over the extended model to measure the distributed secrecy loss.

M. Ibrahim (✉) · R. Kumar
Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011, USA
e-mail: mariami@iastate.edu

R. Kumar
e-mail: rkumar@iastate.edu

M. Ibrahim
Department of Mechatronics Engineering, German Jordanian University, 11180 Amman, Jordan

J. Chen
Idaho National Laboratory, Idaho Falls, USA
e-mail: junchen@iastate.edu

E. Pricop and G. Stamatescu (eds.), *Recent Advances in Systems Safety and Security*, Studies in Systems, Decision and Control 62, DOI 10.1007/978-3-319-32525-5_2

1 Introduction

Growing progress in information and communication technologies has led to growth in eavesdropping and tampering of private communication or behaviors. In contrast to information, behaviors cannot be encrypted, and their *secrecy* can instead be attained through introduction of *covers* that ambiguate secrets in presence of partial observation. Many techniques for hiding secrets based on ambiguation schemes have been proposed as, *Steganography and Watermarking* [5, 16], *Network level Anonymization* [18], and *Software Obfuscation* [9].

Also, various notions of information secrecy have also been explored in literature. For example [1, 8, 21], examine non-interference, requiring that secrets (private variables) do not interfere with or influence the observables (public variables). Non-interference is a logical notion that can only indicate the presence or the absence of interference, but is unable to quantify the level of interference. In contrast, for stochastic systems, the mutual information between the private and public variables can be used to quantify the level of interference, and hence loss of secrecy [21]. Mutual information is only an average case measure, and a worst case measure can also be defined, using for example *min-entropy* [8]. Extension of the notion of non-interference over behaviors (sequences) was explored in [22], requiring that every secret behavior must be masked by a cover behavior so secrets do not uniquely influence the observations.

For probabilistic systems, mutual information can again be used to quantify the level of secrecy loss, and as shown in [3, 12], it can be related to a certain Jensen-Shannon Divergence (JSD) computation, which was first employed in [2] to measure the disparity between the distributions of a secret versus its cover as a way to quantify the secrecy. An approximation algorithm for computing an upper bound of JSD was also provided in [2]. In a similar spirit, Saboori and Hadjicostis [20] considered mutual information between the secret states and the observed behaviors, and required it to be upper bounded. Checking this is undecidable, and Saboori and Hadjicostis [19] proposed a stronger notion, requiring the probability of revealing secrets to remain upper bounded at each time step. In contrast, S_τ-secrecy [11] bounds the probability of revealing secrets over the set of *all* behaviors, as opposed to for each step. S_τ-secrecy can be viewed as a variant of the divergence used in [2]. More related works on secrecy can be found in a recent survey [13].

In this work, we employ the JSD based measure of secrecy loss, and propose a method to compute it for stochastic partially-observed discrete event systems (PODES), under two settings, centralized and distributed. In the centralized setting, the computation of "limiting" JSD measure, quantifying the worst case statistical difference that is defined over arbitrary long observation sequences, is presented. The proposed JSD based quantification for secrecy loss is shown to be equivalent to the mutual information between the distribution over the observations and that over the possible status of system execution (whether secret or cover) [3]. In the distributed collusive setting, there exist multiple observers/attackers that have their own personal observations, and also collude by exchanging their observations over

channels, that introduce delays that are bounded. To compute JSD measure in this setting, we introduce channel models and use those to extend the system model as in [17], capturing own observations as well as the delayed communicated observations. The JSD computation approach of the centralized setting is then employed to the extended model to yield the JSD measure of the distributed collusive setting. Illustrative examples, including one concerning AES (Advanced Encryption Standard), are provided to demonstrate the proposed secrecy loss computation approaches.

2 Notation and Preliminaries

For an event set Σ, define $\overline{\Sigma} := \Sigma \cup \{\varepsilon\}$, where ε denotes "no-event". The set of all finite length event sequences over Σ, including ε is denoted as Σ^*, $\Sigma^+ := \Sigma^* - \{\varepsilon\}$, and Σ^n is the set of event sequences of length $n \in N$. 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. For $s \in \Sigma^*$ and $L \subseteq \Sigma^*$, $L \backslash s := \{t \in \Sigma^* | st \in L\}$ denotes the set of traces in L *after* s.

Stochastic PODES. We can model a stochastic PODES 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 [10], and $\forall x \in X, \sum_{\sigma \in \Sigma} \sum_{x' \in X} \alpha(x, \sigma, x') = 1$. A non-stochastic PODES can be modeled as the same 4-tuple, but by replacing the transition function with $\alpha : X \times \Sigma \times X \to \{0, 1\}$, and a non-stochastic DES 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: $\forall x_i, x_j \in X, s \in \Sigma^*, \sigma \in \Sigma$, $\alpha(x_i, s\sigma, x_j) = \sum_{x_k \in X} \alpha(x_i, s, x_k)\alpha(x_k, \sigma, x_j)$, and $\alpha(x_i, \varepsilon, x_j) = 1$ if $x_i = x_j$ and 0 otherwise.

Define the language generated by G as $L(G) := \{s \in \Sigma^* | \exists x \in X, \alpha(x_0, s, x) > 0\}$. 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 $\alpha_C(x, s, x') > 0$. A 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 which belong to a closed SCC are *recurrent states* and the remaining states (that 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 i jth 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 [24].

Information Theoretic Notations. For a probability distribution p over discrete set A, its entropy is defined as $H(p) = -\sum_{a\in A} p(a)\log p(a)$. For two probability distributions p and q over A, their Kullback-Leibler (KL) divergences denoted as $D_{KL}(p,q)$, is defined as $D_{KL}(p,q) = \sum_{a\in A} p(a)\log\frac{p(a)}{q(a)}$. Given $\lambda_1 > 0$ and $\lambda_2 > 0$ satisfying $\lambda_1 + \lambda_2 = 1$, the Jensen-Shannon Divergence (JSD) between p and q under the weights (λ_1, λ_2), 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 $D(p,q) = H(\lambda_1 p + \lambda_2 q) - \lambda_1 H(p) - \lambda_2 H(q)$ (for more details, refer to [6]). For two probability distributions p over A and q over B, their mutual information is defined as $I(p,q) = \sum_{a\in A, b\in B} Pr(a,b)\log\frac{Pr(a,b)}{p(a)q(b)}$, which can also be equivalently defined as $I(p,q) = H(p) - H(p|q)$, where the conditional entropy $H(p|q)$ is given as $H(p|q) = -\sum_{a\in A} p(a)\sum_{b\in B} Pr(b|a)\log Pr(b|a)$.

3 Illustrative Example: AES Side-Channel Attack

We consider a version of cache side-channel attack that can be used to compromise AES (Advanced Encryption Standard), adopted from [25]. The difference in access times of cache hit versus miss may be used to learn the AES key as described below.

AES is a symmetric crypto-system, which processes data blocks of 16, 24, or 32 bytes, using encryption keys of the same size as data, corresponding to "AES-16", "AES-24", or "AES-32". In what follows below, we consider AES-16 for illustration purposes. For encryption, the plain-text block is converted into the cipher-text block, both viewed as 4×4 array of bytes, in several rounds. The intermediate results of rounds are also of same sizes, and are termed "states". (For AES-16, the number of rounds Nr equals 10 [7, 15].) For setting up the keys for the various rounds, a key expansion algorithm is applied to an initial key $K^{(0)}$, outputting a linear array of 4-byte words, of length $4Nr$, corresponding to the keys $\{K^{(r)}, r = 1, \ldots, N_r\}$ for the future rounds.

Starting from a 16-byte plain-text $P = (p_0, \ldots, p_{15})$, encryption proceeds by computing a 16-byte intermediate state $x^{(r)} = (x_0^{(r)}, \ldots, x_{15}^{(r)})$ at each round r. The initial state $x^{(0)}$ is computed by $x_i^{(0)} = p_i \oplus k_i, i = 0, \ldots, 15$, and the next $Nr - 1$ rounds for $r = 0, \ldots, Nr - 2$ are computed as follows:

$$
\begin{aligned}
(x_0^{(r+1)}, x_1^{(r+1)}, x_2^{(r+1)}, x_3^{(r+1)}) &\leftarrow T_0[x_0^{(r)}] \oplus T_1[x_5^{(r)}] \oplus T_2[x_{10}^{(r)}] \oplus T_3[x_{15}^{(r)}] \oplus K_0^{(r+1)} \\
(x_4^{(r+1)}, x_5^{(r+1)}, x_6^{(r+1)}, x_7^{(r+1)}) &\leftarrow T_0[x_4^{(r)}] \oplus T_1[x_9^{(r)}] \oplus T_2[x_{14}^{(r)}] \oplus T_3[x_3^{(r)}] \oplus K_1^{(r+1)} \\
(x_8^{(r+1)}, x_9^{(r+1)}, x_{10}^{(r+1)}, x_{11}^{(r+1)}) &\leftarrow T_0[x_8^{(r)}] \oplus T_1[x_{13}^{(r)}] \oplus T_2[x_2^{(r)}] \oplus T_3[x_7^{(r)}] \oplus K_2^{(r+1)} \\
(x_{12}^{(r+1)}, x_{13}^{(r+1)}, x_{14}^{(r+1)}, x_{15}^{(r+1)}) &\leftarrow T_0[x_{12}^{(r)}] \oplus T_1[x_1^{(r)}] \oplus T_2[x_6^{(r)}] \oplus T_3[x_{11}^{(r)}] \oplus K_3^{(r+1)},
\end{aligned}
\tag{1}
$$

where the notation $T_n[m]$ denotes the index-m entry of table T_n that is used to store pre-computed transformations of states, involving the operations of substitute-bytes, shift-rows, and mix-columns. The last round is also computed using (1), except that tables $T_0, \ldots, T_3$ are replaced by tables $T_0^{(10)}, \ldots, T_3^{(10)}$, respectively. (The last round does not need the mix-columns operation and so uses different tables.)

An attacker may populate a cache line with an initial state $x_i = p_i \oplus k_i$, generated using a known plain-text p_i and a known key k_i, $i = 0, \ldots, 15$. When the host populates the same cache line with another initial state $x_i' = p_i' \oplus k_i'$, using another plain-text p_i', also known to the attacker, and a key k_i' that is unknown to the attacker, a cache hit, as indicated by a shorter access time, can indicate $x_i = x_i'$, implying $p_i \oplus k_i = p_i' \oplus k_i'$, from which the attacker can infer the unknown key, $k_i' = p_i' \oplus p_i \oplus k_i$. Thus each cache hit, which may be thought of host's cache line interfering with the attacker's cache line, provides an opportunity for an attacker to infer one byte of the key used by a host.

To provide additional protection against this vulnerability, the system may introduce random evictions of the cache. Figure 1a, b show the abstracted versions

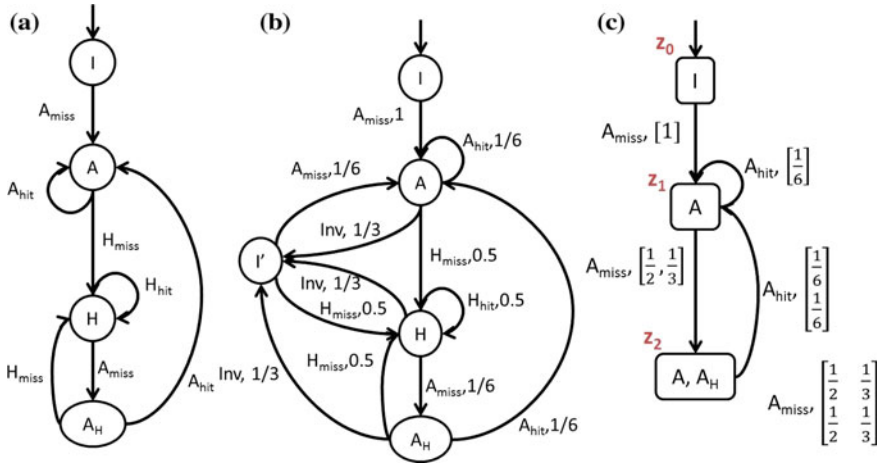

Fig. 1 **a** Cache side-channel attack model with no evictions, **b** cache side-channel attack model with random evictions, **c** observer for the cache side-channel attack with random evictions (reproduced from [3])

of the two cache architectures, with no protection and with added protection, respectively, where the models track the status of an individual cache line. (Similar models track other cache lines.) The 4 states in Fig. 1a are: "A" (occupied by the attacker and of low confidentiality), "H", "A_H" (occupied, respectively, by the host, and the attacker while occupied by the host in the previous step, both of high confidentiality), or "I" (invalid—that has no valid contents from attacker or host, and also of low confidentiality). 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 results in a miss (H_{miss}). The attackers cache hit and miss, A_{hit} and A_{miss} are dually defined. Note that the occurrence of A_{miss} or A_{hit} can be used to infer "H" or "A_H" states, using which one byte of the encryption key can be compromised. However, an attacker can only observe its own cache hits and misses (i.e., A_{hit} and A_{miss} are the only observable events). In Fig. 1b, random cache eviction is introduced by the system to invalidate the data, denoted by "Inv" event. 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. Then, in Fig. 1b, we can view $\{H, A_H\}$ to be the high confidential or "secret" states whereas $\{I, A, I'\}$ to be the low confidential or "cover" states, which present ambiguity against the "secret" states.

4 Quantification of Secrecy Loss in Centralized Setting

In this section, we study secrecy quantification in stochastic PODESs in the presence of a single attacker/observer, having partial observability of system behaviors for revealing sensitive system behaviors, as introduced below.

Secret/non-Secret Behaviors and Refined System. Certain system behaviors may be considered sensitive and hence secret, whereas the remaining behaviors act as covers for the secrets. Letting $L = L(G)$ denote the set of all behaviors (traces) of a stochastic PODES G as introduced in the notation section, suppose $K \subset L$ models the secret behaviors (also called a specification), while the remaining traces in $L - K$ act as its cover. K may be modeled by a deterministic acceptor $R = (Y, \Sigma, \beta, y_0)$ such that $L(R) = K$. By introducing a dump state D in R, and completing its transition function, we can obtain $\overline{R} = (\overline{Y}, \Sigma, \overline{\beta}, y_0)$, where $\overline{Y} = Y \cup D$, and $\forall \bar{y}, \bar{y}' \in \overline{Y}, \sigma \in \Sigma$,

$$\overline{\beta}(\bar{y}, \sigma, \bar{y}') := \begin{cases} \beta(\bar{y}, \sigma, \bar{y}') & \text{if } (\bar{y}, \bar{y}' \in Y) \wedge (\beta(\bar{y}, \sigma, \bar{y}') > 0), \\ 1 & \text{if } [(\bar{y} = \bar{y}' = D) \vee (\bar{y}' = D \wedge \sum_{y \in Y} \beta(\bar{y}, \sigma, y) = 0)]. \end{cases}$$

Then, the system model can be refined with respect to the specification to identify the secret and cover behaviors as *states* in the refined system $G^R = G||\overline{R}$, and is given by $G^R = (X \times \overline{Y}, \Sigma, \gamma, (x_0, y_0))$, where $\forall (x, \bar{y}), (x', \bar{y}') \in X \times \overline{Y}, \sigma \in \Sigma$,

$$\gamma((x,\bar{y}),\sigma,(x',\bar{y}')) := \begin{cases} \alpha(x,\sigma,x') & \text{if}[(\bar{y},\bar{y}' \in Y \wedge \beta(\bar{y},\sigma,\bar{y}') > 0) \vee (\bar{y} = \bar{y}' = D) \\ & \quad \vee (\bar{y}' = D \wedge \sum_{y \in Y} \beta(\bar{y},\sigma,y) = 0)], \\ 0 & \text{otherwise.} \end{cases}$$

The events in Σ executed by the system are observed by an observer (an attacker or an adversary) through an observation mask $M : \overline{\Sigma} \to \overline{\Delta}$, where Δ is the set of observed symbols, and $M(\varepsilon) = \varepsilon$. ($M$ can be extended to Σ^* as follows: $M(\varepsilon) = \varepsilon$ and $\forall s \in \Sigma^*, \sigma \in \overline{\Sigma}, M(s\sigma) = M(s)M(\sigma)$.) The appendix describes the computation of an observer transition structure for G^R that can be used to track its evolution over its observed symbols Δ, and also the associated transition matrices $\{\Theta(\delta) | \delta \in \Delta\}$.

Jensen-Shannon Divergence Based Secrecy Quantification. The statistical difference between the conditional distributions of secrets versus covers over the system observations of a common length, provides a measure of the amount of secrecy leaked by a system. A possible way of measuring difference between two distributions is the JSD (Jensen Shannon Divergence) measure. Here we present a way to compute the JSD measure for stochastic PODESs. The JSD computation can be carried out over the refined system model following the method introduced in [3, 12], which we summarize here.

Given a length-n observation $o \in \Delta^n$, let $p_n(o)$ denote its probability. Then, 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 we can write its entropy as:

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

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

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

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 $\lambda_n^c := Pr(s \in (L-K) \cap M^{-1}(\Delta^n))$ for all $n \in \mathbb{N}$.

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 under the weights $(\lambda_n^s, \lambda_n^c)$, denoted $D(p_n^s, p_n^c) = H(\lambda_n^s p_n^s + \lambda_n^c p_n^c) - \lambda_n^s H(p_n^s) - \lambda_n^c H(p_n^c)$.

The following theorem from [3] shows that the JSD measure is indeed a useful measure of information revealed, as it equals the mutual information between the observations p_n and the status (whether secret or cover) of system executions. This status can be captured by a bi-valued random variable Λ_n, defined for each $n \in \mathbb{N}$, such that $Pr(\Lambda_n = s) = \lambda_n^s$ and $Pr(\Lambda_n = c) = \lambda_n^c$.

Theorem 1 ([3]). *The JSD between p_n^s and p_n^c equals the mutual information between Λ_n and p_n, i.e.,*

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

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

In order to compute JSD, we need to first compute the state-distribution of the observer, following each observation. Each observation $o \in \Delta^*$ results in a conditional state distribution $\pi(o)$, which can be computed recursively as follows: for any $o \in \Delta^*, \delta \in \Delta$: $\pi(\varepsilon) = \pi_0$ and $\pi(o\delta) = \frac{\pi(o)\times\Theta(\delta)}{||\pi(o)\times\Theta(\delta)||}$ [4], where π_0 is the initial state distribution, whereas the computation of transition matrix $\Theta(\delta)$ is given in the appendix. 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 the conditional state distribution is π. For a state distribution π, define the following notations:

$$\lambda^{s|\pi} := \sum_{\delta\in\Delta} \pi\Theta(\delta)\mathscr{I}^s, \quad \lambda^{c|\pi} := \sum_{\delta\in\Delta} \pi\Theta(\delta)\mathscr{I}^c$$
$$p^{s|\pi}(\delta) := \frac{\pi\Theta(\delta)\mathscr{I}^s}{\lambda^{s|\pi}}, \quad p^{c|\pi}(\delta) := \frac{\pi\Theta(\delta)\mathscr{I}^c}{\lambda^{c|\pi}},$$

where $\mathscr{I}^s$ and $\mathscr{I}^c$ denote indicator column vectors of same size as number of states, with binary entries to identify the secret versus cover states (states reached by traces in K vs. $L - K$). Then, as shown in Lemma 4 of [3],

$$D(p_n^s, p_n^c) = H(\{\lambda_n^s, \lambda_n^c\}) + \sum_{\pi\in\Pi} P_{n-1}(\pi)\Big[-H(\{\lambda^{s|\pi}, \lambda^{c|\pi}\}) + D(p^{s|\pi}, p^{c|\pi})\Big]. \quad (2)$$

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. See for example [14] for a condition under which such a convergence is guaranteed.

For an observer *Obs*, the computation of $\lim_{n\to\infty} D(p_n^s, p_n^c)$ using (2), requires the computation of $\lim_{n\to\infty} P_{n-1}(\pi)$ which can be accomplished with the help of an observer introduced in [3, 12]. The observer tracks the possible system states following each observation, and also allows the computation of the corresponding state distribution. We let *Obs* be an observer automaton with state set $Z \subseteq 2^{X\times\overline{Y}}$, so that each node $z \in Z$ of the observer is a subset of the refined system states, i.e., $z \subseteq (X, \overline{Y})$, 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., $z' = \{(x', \bar{y}') \in X \times \overline{Y} : \exists (x, \bar{y}) \in z,\ L_{G^R}((x, \bar{y}), \delta, (x', \bar{y}')) \neq \emptyset\}$. Associated with this transition is the transition probability matrix $\Theta_{z,\delta,z'}$ of size $|z|$ by $|z'|$ (a submatrix of $\Theta(\delta)$ matrix given in the appendix), whose *ijth* element is $\theta_{i,\delta,j}$, which is the transition probability from *ith* element $(x, \bar{y})$ of z to *jth* element $(x', \bar{y}')$ of z' while producing the observation δ, and equals $\alpha(L_{G^R}((x, \bar{y}), \delta, (x', \bar{y}')))$.

Example 1 Consider the system, specification and refinement models of Fig. 2a–c, respectively, where $M(u) = \varepsilon$, $M(a) = a$ and $M(b) = b$. Then, the corresponding observer *Obs* is given in Fig. 2d, where each state in observer is a subset of states of the refined-system G^R, and transitions are on observed events that are labeled by their occurrence transition probability matrices.

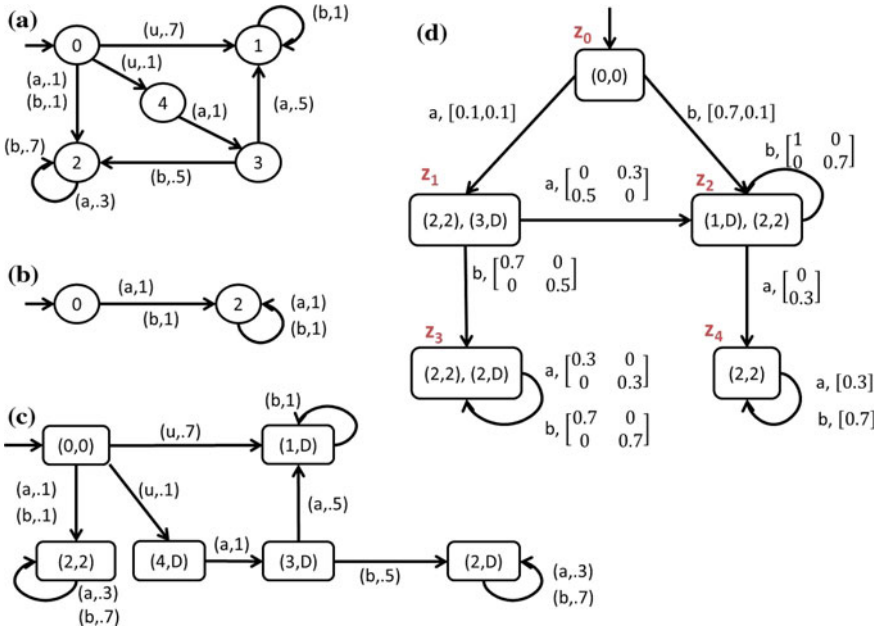

Fig. 2 **a** System model G, **b** specification for secrets, R, **c** refined system model G^R, **d** observer model (reproduced from [3])

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 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 $o \in \Delta^*, \delta \in \Delta$: $\pi(\varepsilon) = 1$ and $\pi(o\delta) = \frac{\pi(o) \times \Theta_{z_o,\delta,z_{o\delta}}}{||\pi(o) \times \Theta_{z_o,\delta,z_{o\delta}}||}$, 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 normalizations. In case of steady-state, those two distributions will be the same, namely, a fixed point of that function. The following assumption is made as in [3, 12].

Assumption 1 ([3, 12]) 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 as shown in [3, 12], the following procedure computes the worst-case loss of secrecy $\lim_{n\to\infty} D(p_n^s, p_n^c)$, under Assumption 1.

1. Construct a $\left(\sum_z |z|\right) \times \left(\sum_z |z|\right)$ square matrix $\tilde{\Theta}$, whose *ijth* block is the $|z_i| \times |z_j|$ matrix $\sum_\delta \Theta_{z_i,\delta,z_j}$. Compute the fix point distribution associated with $\tilde{\Theta}$ by solving $\pi^* = \pi^* \tilde{\Theta}$, 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 *recurrent* if $p(z_i) > 0$. Also note that for each $z \in Z$, exists a sufficiently large N such that $p(z) = \sum_{o \in \Delta^N : o \text{ reaches } z} p_N(o)$. In other words, $p(z)$ computes the probability of all sufficiently long observations that reach the observer state z.
2. Obtain λ^s as the summation of the elements of π^* corresponding to the secret states, i.e., $\lambda^s := \pi^* \mathcal{I}^s$, and $\lambda^c = 1 - \lambda^s$.
3. For a set of recurrent nodes $\{z_1, z_2, \ldots, z_n\}$ that form a SCC, 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, 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 say as $\{\{\pi^*_{z_1,k}, \ldots, \pi^*_{z_n,k}\}, k \in \mathbb{N}\}$. 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 $p(z,k) := Pr[\{o | o \text{ reaches } z \text{ and } \pi(o) = \pi^*_{z,k}\}]$.

4. Let $\mathscr{I}^s_{z'}$ and $\mathscr{I}^c_{z'}$ 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:

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

5. Then, applying (2), the JSD between p^s_n and p^c_n when $n \to \infty$ is given by:

$$\lim_{n\to\infty} D(p^s_n, p^c_n) = H(\{\lambda^s, \lambda^c\}) + \sum_{z:z \text{ is recurrent}} \sum_{k\in\mathbb{N}} p(z,k)\Big[-H(\{\lambda^{s|\pi^*_{z,k}}, \lambda^{c|\pi^*_{z,k}}\}) + D(p^{s|\pi^*_{z,k}}, p^{c|\pi^*_{z,k}})\Big]. \tag{3}$$

(Note when the set of steady state distributions is unique, then in that case, $k = 1$ and we have: $p(z,k) = p(z)$ in (3) above.)

Example 2 We revisit Example 1. Then based on *Obs* of Fig. 2d, the following computation illustrates the steps of JSD computation.

1. $\sum_z |z| = 8$ and so $\tilde{\Theta}$ is a 8×8 matrix with entries: $\tilde{\Theta}(1,2) = \tilde{\Theta}(1,3) = \tilde{\Theta}(1,5) = 0.1$, $\tilde{\Theta}(3,4) = \tilde{\Theta}(3,7) = 0.5$, $\tilde{\Theta}(1,4) = \tilde{\Theta}(2,6) = \tilde{\Theta}(5,5) = 0.7$, $\tilde{\Theta}(2,5) = \tilde{\Theta}(5,8) = 0.3$, $\tilde{\Theta}(4,4) = \tilde{\Theta}(6,6) = \tilde{\Theta}(7,7) = \tilde{\Theta}(8,8) = 1$, and zeros elsewhere. Then, $\pi^* = [0 \quad 0 \quad 0 \quad 0.75 \quad 0 \quad 0.07 \quad 0.05 \quad 0.13]$. Therefore, $p(z_0) = p(z_1) = 0$, $p(z_2) = 0.75$, $p(z_3) = 0.12$ and $p(z_4) = 0.13$.
2. Here $\mathscr{I}^s = [1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1]^T$, $\mathscr{I}^c = [0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0]^T$. And so, $\lambda^s = 0.2$ and $\lambda^c = 0.8$.
3. Here z_2, z_3 and z_4 are recurrent nodes, and each of them forms a SCC. We have $\pi^*_{z_2} = [1 \quad 0]$, $\pi^*_{z_4} = [1]$, and while there are multiple solutions to the equation set $\pi^*_{z_3} = \frac{\pi^*_{z_3}\Theta_{z_3,a,z_3}}{\pi^*_{z_3}\Theta_{z_3,a,z_3}}$ and $\pi^*_{z_3} = \frac{\pi^*_{z_3}\Theta_{z_3,b,z_3}}{\pi^*_{z_3}\Theta_{z_3,b,z_3}}$, only $\pi^*_{z_3} = [0.5833 \quad 0.4167]$ is reachable. Thus, each set of recurrent nodes is a singleton set, and each with a unique fixed-point distribution. Therefore, for each recurrent node z, $p(z,k) = p(z)$.
4. Here $\mathscr{I}^s_{z_2} = [0 \quad 1]^T$, $\mathscr{I}^c_{z_2} = [1 \quad 0]^T$, $\mathscr{I}^s_{z_3} = [1 \quad 0]^T$, $\mathscr{I}^c_{z_3} = [0 \quad 1]^T$, $\mathscr{I}^s_{z_4} = [1]^T$ and $\mathscr{I}^c_{z_4} = [0]^T$. For z_2 and $\pi^*_{z_2}$, $\lambda^{s|\pi^*_{z_2}} = 0, \lambda^{c|\pi^*_{z_2}} = 1$, $p^{c|\pi^*_{z_2}}(b) = \frac{\pi^*_{z_2}\Theta_{z_2,b,z_2}\mathscr{I}^c_{z_2}}{\lambda^{c|\pi^*_{z_2}}} = 1$, $p^{s|\pi^*_{z_2}}(a) = p^{c|\pi^*_{z_2}}(a) = p^{s|\pi^*_{z_2}}(b) = 0$. For z_3 and $\pi^*_{z_3}$, $\lambda^{s|\pi^*_{z_3}} = 0.5833, \lambda^{c|\pi^*_{z_3}} = 0.4167$, $p^{s|\pi^*_{z_3}}(a) = \frac{\pi^*_{z_3}\Theta_{z_3,a,z_3}\mathscr{I}^s_{z_3}}{\lambda^{s|\pi^*_{z_3}}} = 0.3$, $p^{s|\pi^*_{z_3}}(b) = \frac{\pi^*_{z_3}\Theta_{z_3,b,z_3}\mathscr{I}^s_{z_3}}{\lambda^{s|\pi^*_{z_3}}} = 0.7$, $p^{c|\pi^*_{z_3}}(a) = \frac{\pi^*_{z_3}\Theta_{z_3,a,z_3}\mathscr{I}^c_{z_3}}{\lambda^{c|\pi^*_{z_3}}} = 0.3$, $p^{c|\pi^*_{z_3}}(b) = \frac{\pi^*_{z_3}\Theta_{z_3,b,z_3}\mathscr{I}^c_{z_3}}{\lambda^{c|\pi^*_{z_3}}} = 0.7$.

For z_4 and $\pi^*_{z_4}$, $\lambda^{s|\pi^*_{z_4}} = 1, \lambda^{c|\pi^*_{z_4}} = 0$, $p^{s|\pi^*_{z_4}}(a) = \frac{\pi^*_{z_4} \Theta_{z_4,a,z_4} \mathscr{I}^s_{z_4}}{\lambda^{s|\pi^*_{z_4}}} = 0.3$, $p^{s|\pi^*_{z_4}}(b) = \frac{\pi^*_{z_4} \Theta_{z_4,b,z_4} \mathscr{I}^s_{z_4}}{\lambda^{s|\pi^*_{z_4}}} = 0.7$, $p^{c|\pi^*_{z_4}}(a) = p^{c|\pi^*_{z_4}}(b) = 0$.

5. Then, we have

$$\begin{aligned}\lim_{n\to\infty} D(p^s_n, p^c_n) &= H(\{\lambda^s, \lambda^c\}) \\ &\quad + \sum_{z:z \text{ is recurrent}} p(z)[-H(\{\lambda^{s|\pi^*_z}, \lambda^{c|\pi^*_z}\}) + D(p^{s|\pi^*_z}, p^{c|\pi^*_z})] \\ &= \mathbf{0.6043}.\end{aligned}$$

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

Application to Cache Side-Channel Attack. For the cache side-channel attack model of Fig. 1b, the observer model is given in Fig. 1c. It can be computed that $p(z_1) = 1/6$, $p(z_2) = 5/6$, $\pi^*_{z_1} = [1]$, $\pi^*_{z_2} = [0.6 \quad 0.4]$, $\lambda_s = 1/3$ and $\lambda_c = 2/3$. From which, the limiting divergence $\lim_{n\to\infty} D(p^s_n, p^c_n) = \mathbf{0}$, meaning that no amount of secrecy could be leaked through the side-channel if the cache line is periodically evicted by the processor.

5 Quantification of Distributed Secrecy Loss in Stochastic PODESs Under Bounded-Delay Communications

We now extend the analysis of previous section to study the secrecy quantification in stochastic PODESs in the presence of distributed collusive attackers/observers, each with its own local partial observability, and where the local observers collude and exchange their observations over communication channels with bounded delays, to be able to infer more about the system secrets.

***d*-Delaying&Masking Communication Channel**. Figure 3a shows the architecture of a system with distributed observers/attackers, where it is assumed for simplicity and without loss of any generality that there are two local observers at two local sites $I = \{1, 2\}$. Each site has three modules [17]: (i) observation mask $M_i : \overline{\Sigma} \to \overline{\Delta}_i$, where Δ_i is the set of locally observed symbols and $M_i(\varepsilon) = \varepsilon$ (M_i can be extended to Σ^* as follows: $M_i(\varepsilon) = \varepsilon$, and $\forall s \in \Sigma^*, \sigma \in \overline{\Sigma}, M_i(s\sigma) = M_i(s)M_i(\sigma)$), (ii) communication channels $C^{(d)}_{ij}, j \neq i, i, j \in I$, which are lossless and order-preserving, but introduce delays bounded by d, and (iii) observer Obs_i, that tracks the system "information-state" following the arrival of its local observations and the communicated observations received from other sites $j \in I, j \neq i$.

The communication channel is a "*delay-block*" with d-bounded communication delay that holds the transmitted information in First-In-First-Out (*FIFO*) manner for at most d delay steps. Accordingly, since there can be at most d events executed by

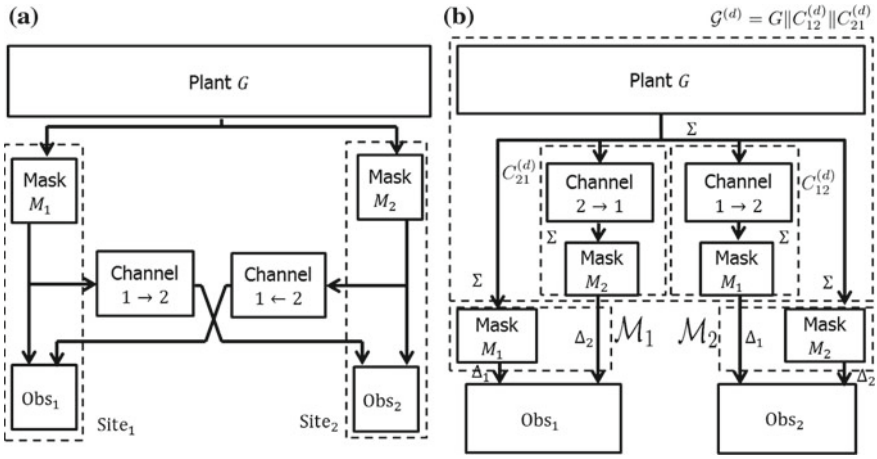

Fig. 3 **a** Distributed secrecy system architecture to **b** equivalent system architecture (reproduced from [17])

system G between the transmission and the reception of a message on a channel, the channel has a maximum queue length $d+1$. Also, the channel queue evolves whenever a system event occurs, or a transmitted observation is delivered to a destination observer, where such arrival and departure events occur asynchronously. Accordingly, the d-delaying&masking non-stochastic channel model from site-i to site-j $(i \neq j, i, j \in I)$ is of the form, $C_{ij}^{(d)} = (Q_{ij}^{(d)}, \Sigma \cup \overline{\Delta}_i, \beta_{ij}^{(d)}, q_0)$, with the elements as follows. $Q_{ij}^{(d)} \subseteq \Sigma^*$ denotes the set of states, which are the event traces executed in the system but their observed values pending to be delivered at the destination. For $q \in Q_{ij}^{(d)}$, it holds that $|q| \leq d+1$. $\Sigma \cup \overline{\Delta}_i$ is the event set of $C_{ij}^{(d)}$, where Σ is its set of input events and $\overline{\Delta}_i$ is its set of output events. Without loss of generality, we assume that $\Sigma \cap \overline{\Delta}_i = \emptyset$, and $\Delta_i \cap \Delta_j = \emptyset, (j \neq i)$ (otherwise, we can simply rename some of the symbols). $q_0 = \varepsilon$ is the initial state, whereas the transition function $\beta_{ij}^{(d)}$ is defined as follows:

1. "Arrival" due to an event execution in the system: $\forall q \in Q_{ij}^{(d)}, \forall \sigma \in \Sigma$, if $|q| \leq d$, then $\beta_{ij}^{(d)}(q, \sigma) = q\sigma$,
2. "Departure" due to a reception at the destination observer: $\forall q \in Q_{ij}^{(d)}, \forall \delta_i \in \overline{\Delta}_i$, if $M_i(head(q)) = \delta_i$, then $\beta_{ij}^{(d)}(q, \delta_i) = q \backslash head(q)$,
3. Undefined, otherwise,

where $head(q)$ is the first event in trace q, and the after operator "\" in $q \backslash head(q)$ returns the trace after removing the initial event $head(q)$ from the trace q.

Example 3 A system model G is shown in Fig. 4a, with $L(G) = a^+ \cup ba^* \cup uba^+$. Suppose the observation masks of two local sites are defined as follows:

- $M_1(a) = a'$, $M_1(b) = M_1(u) = \varepsilon$, and
- $M_2(b) = b'$, $M_2(a) = M_2(u) = \varepsilon$.

For delay $d = 0$, Fig. 4b shows the model $C_{12}^{(0)}$, and for *delay* $d = 1$, Fig. 4c, d show the models $C_{12}^{(1)}$ and $C_{21}^{(1)}$, respectively. If we follow the trace bab' in $C_{21}^{(1)}$, the states ε, b, ba and a are traversed sequentially. This corresponds to the situation in which site-2 sends out its observation b' to site-1 following the execution of ba in the system, whereas the observation of event a is pending to be received at site-1.

Next, since the operations of masking and delaying can be interchanged, the behaviors under the schematic of Fig. 3a are equivalent to those of Fig. 3b. Then, it is clear that the distributed setting of Fig. 3a can be converted to a decentralized setting of Fig. 3b, having an extended system $\mathscr{G}^{(d)}$ and local observers having the extended observation masks $\{\mathscr{M}_i\}$, defined below. The extended system is given by $\mathscr{G}^{(d)} = G||_{i,j\in I, i\neq j} C_{ij}^{(d)}$, whereas the extended system model $\mathscr{G}_i$ at site-i $(i \in I)$ includes the system model and only the incoming channel models: $\mathscr{G}_i = G||_{j\in I-\{i\}} C_{ji}^{(d)}$. The *extended system* $\mathscr{G}_i$ "generates" events in $\Sigma \cup_{j\neq i} \Delta_j$, which are observed by site-i observer Obs_i through an extended observation mask $\mathscr{M}_i : \overline{\Sigma} \cup_{j\in I-\{i\}} \overline{\Delta}_j \to \overline{\Delta} = \cup_{i\in I} \overline{\Delta}_i$. $\mathscr{M}_i$ acts the same as M_i for events in Σ, whereas it is an identity mask for events in Δ_j $(j \neq i)$. Formally, it is defined as follows:

$$\mathscr{M}_i(\sigma) := \begin{cases} M_i(\sigma), & \sigma \in \Sigma, \\ \sigma, & \sigma \in \Delta_j \ (j \neq i). \end{cases} \tag{4}$$

The extended system model at site-i $(i \in I)$ can be refined with respect to the specification to identify the secret and cover behaviors as *states* in the refined system, and is given by $\mathscr{G}_i^R = G||_{j\in I-\{i\}} C_{ji}^{(d)} || \overline{R}$.

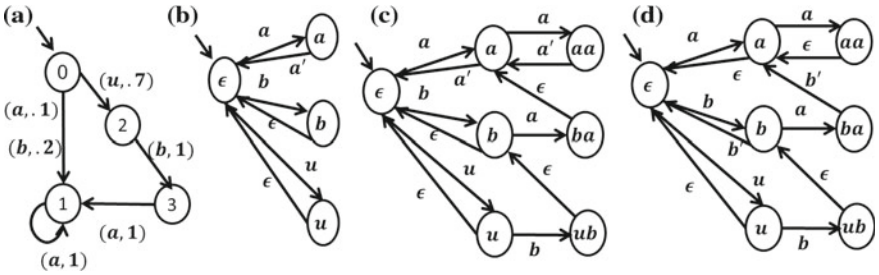

Fig. 4 **a** Stochastic PODES G, **b** $C_{12}^{(0)}$, **c** $C_{12}^{(1)}$, **d** $C_{21}^{(1)}$

Next, we assign probabilities to transitions in $\mathscr{G}_i^R$ as follows. For each state in $\mathscr{G}_i^R$, the transition is either one of the system events, or at most one of channel j ($j \neq i$) events (either arrival or departure of that channel). Suppose at a system $\mathscr{G}_i^R$ state, with vector of all incoming channel lengths $\mathbf{k}$, the system event is picked with probability $p_{\mathbf{k}}^0$, and suppose the channel j ($j \neq i$) event can occur with probability p_k^j such that, $p_{\mathbf{k}}^0 + \sum_{j \neq i} p_{\mathbf{k}}^j = 1$. We also require that when all channels are empty ($\mathbf{k} = \mathbf{0}$), $p_{\mathbf{k}}^0 = 1$ (so no channel output can occur when channels are empty), when all channels are full ($\mathbf{k} = \overrightarrow{d+1}$), $p_{\mathbf{k}}^0 = 0$ (so no channel input can occur when channels are full), and if channel j has higher queue length than channel j' ($\mathbf{k_j} \geq \mathbf{k'_j}$), then it can be expected that $p_{\mathbf{k}}^j \geq p_{\mathbf{k}}^{j'}$ (channel j event is more likely than channel j' event when channel j has more number of pending observations). With this choice of selection probability of events, refined extended system model is given by $\mathscr{G}_i^R = (X \times (\Pi_{j \neq i} Q_{ji}^{(d)}) \times \overline{Y}, \Sigma \cup_{j \neq i} \overline{\Delta}_j, \gamma, (x_0, \mathbf{q}_0, y_0))$, where $\overline{Y} = Y \cup \{D\}$, and $\forall (x, \mathbf{q}, \bar{y})$, $(x', \mathbf{q}', \bar{y}') \in X \times (\Pi_{j \neq i} Q_{ji}^{(d)}) \times \overline{Y}$, $\sigma \in \Sigma \cup_{j \neq i} \overline{\Delta}_j$,

$$\gamma((x, \mathbf{q}, \bar{y}), \sigma, (x', \mathbf{q}', \bar{y}')) = \begin{cases} \alpha(x, \sigma, x') \times p_{\mathbf{k}}^0 & \text{if } \sigma \in \Sigma, \\ p_{\mathbf{k}}^j & \text{if } \sigma \in \cup_{j \neq i} \overline{\Delta}_j, \end{cases}$$

if the following holds:

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

and otherwise, $\gamma((x, \mathbf{q}, \bar{y}), \sigma, (x', \mathbf{q}', \bar{y}')) = 0$.

The computation of an observer transition structure for $\mathscr{G}_i^R$ and the associated transition matrices $\{\Theta(\delta) | \delta \in \Delta\}$, is exactly the same as in the centralized setting, as is described in the appendix.

Example 4 Continuing Example 3, suppose the delay bound $d = 1$, so there are three possibilities for the length of the only channel, $\mathbf{k} = \{0, 1, 2\}$. Let $p_0^0 = 1, p_1^0 = 0.5, p_2^0 = 0$ (implying $p_0^2 = 1 - p_0^0 = 0$, $p_1^2 = 1 - p_1^0 = 0.5$, $p_2^2 = 1 - p_2^0 = 1$). Figure 5a shows the extended system model $\mathscr{G}_1$ at site-1. Suppose R is given in Fig. 5b, i.e., $K = L(R) = a^+ \cup ba^*$. Then, the refinement G_1^R is shown in Fig. 5c. So for example, at the initial state $(0, \varepsilon, 0)$, the channel is empty, and no channel events occur at this state ($p_0^2 = 0$ while $p_0^0 = 1$). Then, for any system event $\sigma \in \Sigma$, $\gamma((0, \varepsilon, 0), u, (2, u, D)) = \alpha(0, u, 2) \times p_0^0 = 0.7 \times 1 = 0.7$, $\gamma((0, \varepsilon, 0), b, (1, b, 1)) = \alpha(0, b, 1) \times p_0^0 = 0.2 \times 1 = 0.2$, and $\gamma((0, \varepsilon, 0), a, (1, a, 1)) = \alpha(0, a, 1) \times p_0^0 = 0.1 \times 1 = 0.1$. Whereas, at state $(2, u, D)$, there is observation u queued up in the channel. Thus, either the system can execute a new event $b \in \Sigma$, with probability $\gamma((2, u, D), b, (3, ub, D)) = \alpha(2, b, 3) \times p_1^0 = 1 \times p_1^0 = 0.5$, or a channel event can occur, with probability $\gamma((2, u, D), \varepsilon, (2, \varepsilon, D)) = p_1^2 = 0.5$. The remaining state

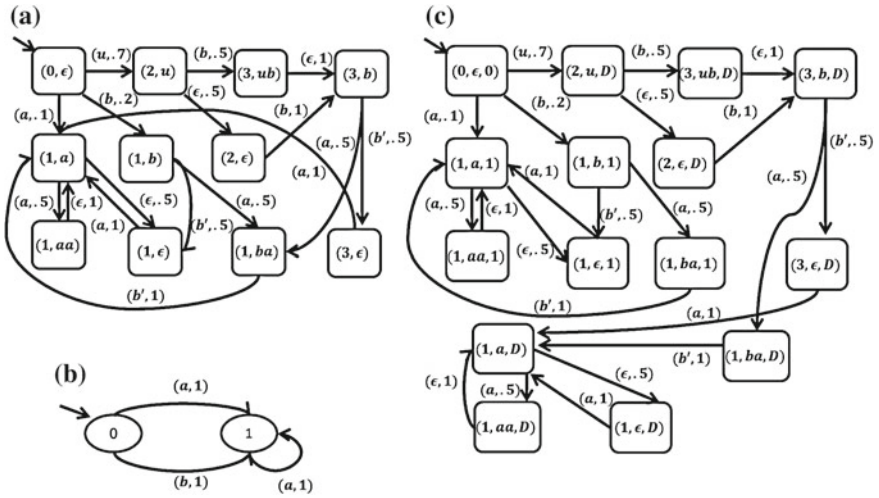

Fig. 5 **a** Extended system model $\mathscr{G}_1$ at site-1, **b** specification for secrets, R, **c** refined system model $\mathscr{G}_1^R$

transitions can be computed similarly. The models $\mathscr{G}_2$ and $\mathscr{G}_2^R$ at site-2 can be generated in a manner similar to $\mathscr{G}_1$ and $\mathscr{G}_1^R$, respectively.

In Sect. 4, we presented a way to compute JSD-based measure of secrecy loss for stochastic PODES when there is a single observer. To compute the secrecy loss in the distributed setting, resulting from the aggregated observations at any site-i $(i \in I)$, which include it's own immediate observations and the delayed communicated observations from other distributed sites, the JSD computation can be carried out over the refined extended system model $\mathscr{G}_i^R$, following the method introduced in Sect. 4. The example below illustrates the extended observer structure and the corresponding JSD based secrecy loss computation in a distributed collusive setting, respectively.

Example 5 Consider the refined extended system model of Fig. 5c at site-1 where $\mathscr{M}_1(a) = a'$, $\mathscr{M}_1(b) = \mathscr{M}_1(u) = \varepsilon$, while the extended mask function is the identity function over the received observations, $\Delta_2 = \{b'\}$. Then, Fig. 6a shows the extended observer Obs_1.

Then, based on Obs_1, the following computation illustrates the steps of JSD computation at site-1.

1. $\sum_z |z| = 14$ and so $\tilde{\Theta}$ is a 14×14 matrix with entries:
$\tilde{\Theta}(1,2) = \tilde{\Theta}(1,3) = \tilde{\Theta}(1,5) = 0.1, \quad \tilde{\Theta}(1,4) = \tilde{\Theta}(1,6) = 0.35, \quad \tilde{\Theta}(2,7) = \tilde{\Theta}(2,8) = \tilde{\Theta}(7,7) = \tilde{\Theta}(7,8) = \quad \tilde{\Theta}(8,7) = \tilde{\Theta}(8,8) = \tilde{\Theta}(9,11) = \tilde{\Theta}(9,12) = \tilde{\Theta}(10,13) = \tilde{\Theta}(10,14) = \tilde{\Theta}(11,11) = \tilde{\Theta}(11,12) = \tilde{\Theta}(12,11) = \tilde{\Theta}(12,12) =$

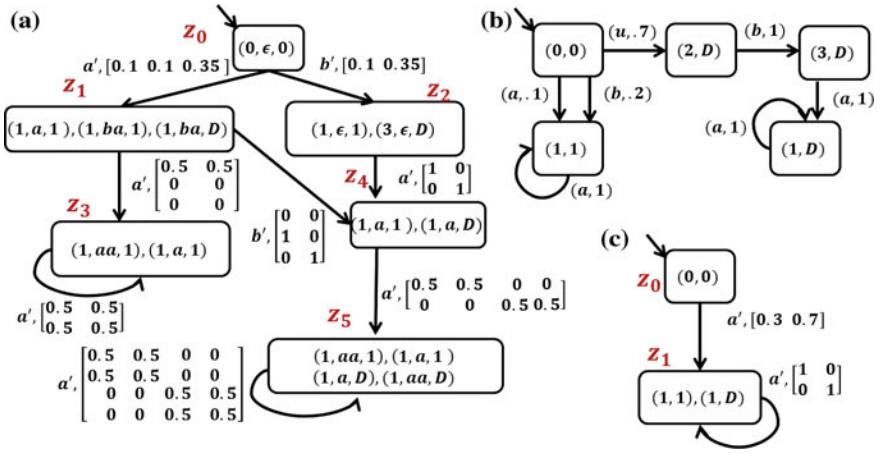

Fig. 6 **a** Observer Obs_1 for the system of Fig. 5c, **b** model G^R for system of Fig. 4a under no collusion, **c** observer under no collusion

$\tilde{\Theta}(13,13) = \tilde{\Theta}(13,14) = \tilde{\Theta}(14,13) = \tilde{\Theta}(14,14) = 0.5$, $\tilde{\Theta}(3,9) = \tilde{\Theta}(4,10) = \tilde{\Theta}(5,9) = \tilde{\Theta}(6,10) = 1$ and zeros elsewhere.
Then, $\pi^* = [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0.05\ \ 0.05\ \ 0\ \ 0\ \ 0.1\ \ 0.1\ \ 0.35\ \ 0.35]$. Therefore, $p(z_0) = p(z_1) = p(z_2) = 0$, $p(z_3) = 0.1$, $p(z_4) = 0$, and $p(z_5) = 0.9$.

2. Here $\mathscr{I}^s = [1\ \ 1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 1\ \ 1\ \ 0\ \ 0]^T$, $\mathscr{I}^c = [0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 1\ \ 1]^T$. And so $\lambda^s = 0.3$ and $\lambda^c = 0.7$.
3. Here z_3, and z_5 are recurrent nodes, and each of them forms a SCC. We have $\pi^*_{z_3} = [0.5\ \ 0.5]$, and while there are multiple solutions to the equation set $\pi^*_{z_5} = \frac{\pi^*_{z_5}\Theta_{z_5,a',z_5}}{||\pi^*_{z_5}\Theta_{z_5,a',z_5}||}$, only $\pi^*_{z_5} = [0.11\ \ 0.11\ \ 0.39\ \ 0.39]$ is reachable. Thus, each set of recurrent nodes is a singleton set, and each with a unique fixed-point distribution. Therefore, for each recurrent node z, $p(z,k) = p(z)$.
4. Here $\mathscr{I}^s_{z_3} = [1\ \ 1]^T$, $\mathscr{I}^c_{z_3} = [0\ \ 0]^T$, $\mathscr{I}^s_{z_5} = [1\ \ 1\ \ 0\ \ 0]^T$, $\mathscr{I}^c_{z_5} = [0\ \ 0\ \ 1\ \ 1]^T$. For z_3 and $\pi^*_{z_3}$, $\lambda^{s|\pi^*_{z_3}} = 1, \lambda^{c|\pi^*_{z_3}} = 0$, $p^{s|\pi^*_{z_3}}(a') = \frac{\pi^*_{z_3}\Theta_{z_3,a',z_3}\mathscr{I}^s_{z_3}}{\lambda^{s|\pi^*_{z_3}}} = 1$, $p^{s|\pi^*_{z_3}}(b') = p^{c|\pi^*_{z_3}}(b') = p^{c|\pi^*_{z_3}}(a') = 0$. For z_5 and $\pi^*_{z_5}$, $\lambda^{s|\pi^*_{z_5}} = 0.22, \lambda^{c|\pi^*_{z_5}} = 0.78$, $p^{s|\pi^*_{z_5}}(a') = \frac{\pi^*_{z_5}\Theta_{z_5,a',z_5}I^s_{z_5}}{\lambda^{s|\pi^*_{z_5}}} = 1$, $p^{c|\pi^*_{z_5}}(a') = \frac{\pi^*_{z_5}\Theta_{z_5,a',z_5}I^c_{z_5}}{\lambda^{c|\pi^*_{z_5}}} = 1$, $p^{s|\pi^*_{z_5}}(b') = p^{c|\pi^*_{z_5}}(b') = 0$.

5. Then, we have

$$\lim_{n\to\infty} D_1(p_n^s, p_n^c) = H(\{\lambda^s, \lambda^c\}) + \sum_{z:z \text{ is recurrent}} p(z)[-H(\{\lambda^{s|\pi_z^*}, \lambda^{c|\pi_z^*}\}) + D_1(p^{s|\pi_z^*}, p^{c|\pi_z^*})] = \mathbf{0.197}.$$

Note this happens to be the same as JSD measure of secrecy loss at site-2.

In contrast, when there is no collusion among observers (so there is no communication among the two sites), Fig. 6b, c show, respectively, the refined system G^R (no incoming channels and so identical refined model at all sites) and the corresponding site-1 observer structure. The JSD value, computed in same manner as above but with respect to the observer structure of Fig. 6c, is simply **Zero**, i.e., no amount of secrets is revealed under no collusion. This is because for every observation, the probability of it coming from secrets in K vs from covers in $L - K$ is exactly the same.

6 Conclusion

In this chapter, we presented information theoretic measure for secrecy loss quantification in PODESs in both centralized versus distributed collusive settings, in the presence of a single attacker/observer versus multiple attackers/observers exchanging their observations, respectively. The statistical difference, in the form of the Jensen-Shannon Divergence, between the influence of secrets versus covers on the observations, is employed to quantify the loss of secrecy. It is shown that this JSD measure is equivalent to the mutual information between the distribution over the possible observations versus that over the possible status of system execution (whether secret or cover). An observer structure is formed and used to aide the computation of JSD in the limit as the length of the observation approaches infinity to quantify the worst case loss of secrecy. In distributed collusive setting, channel models are introduced to extend the system model to capture the effect of exchange of observations, and the JSD computation of the centralized case is applied over the extended model to arrive at the measure for secrecy loss. Future work will involve developing a software tool for JSD computation, and performing application studies. Knowing the JSD value can help an engineer to perform secrecy analysis of a system, and revisit the system design to make it improve its level of secrecy as needed.

Acknowledgments This research was supported in part by Security and Software Engineering Research Center (S2ERC), and the National Science Foundation under the grants NSF-CCF-1331390 and NSF-ECCS 1509420.

Appendix

In this appendix, we describe the computation of an observer transition structure that can be used to track the evolution of G^R over its observed symbols Δ, and the associated transition matrices $\{\Theta(\delta)|\delta \in \Delta\}$. Given the refined system model G^R, and its observation mask $M : \overline{\Sigma} \to \overline{\Delta}$, 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 $L_{G^R}((x, \bar{y}), \delta, (x', \bar{y}')) := \{s \in \Sigma^* | s = u\sigma,\ M(u) = \varepsilon, M(\sigma) = \delta,\ \gamma((x, \bar{y}), s, (x', \bar{y}')) > 0\}$. Define its probability, $\alpha(L_{G^R}((x, \bar{y}), \delta, (x', \bar{y}'))) := \sum_{s \in L_{G^R}((x,\bar{y}),\delta,(x',\bar{y}'))} \gamma((x, \bar{y}), s, (x', \bar{y}'))$, and denote it as $\theta_{(x,\bar{y}),\delta,(x',\bar{y}')}$. Also, 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, letting $i = (x, \bar{y})$ and $j = (x', \bar{y}')$, $\theta_{i,\delta,j} = \sum_k \lambda_{ik}\theta_{k,\delta,j} + \sum_{\sigma \in \Sigma : M(\sigma) = \delta} (i, \sigma, j)$, 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 RHS corresponds to transitioning in exactly one step [3, 12]. Thus, for each $\delta \in \Delta$, all the probabilities $\{\theta_{i,\delta,j} | i, j \in X \times \overline{Y}\}$ can be found by solving the following matrix equation [23]: $\Theta(\delta) = \Lambda\Theta(\delta) + \Gamma(\delta)$, where $\Theta(\delta)$, Λ and $\Gamma(\delta)$ are all $|X \times \overline{Y}| \times |X \times \overline{Y}|$ 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.

References

1. Backes, M., Köpf, B., Rybalchenko, A.: Automatic discovery and quantification of information leaks. In: Proceedings of 30th IEEE Symposium on Security and Privacy, pp. 141–153, Washington, DC, 2009
2. Bryans, J., Koutny, M., Mu, C.: Towards quantitative analysis of opacity. Technical Reports Series, Newcastle University (2011)
3. Chen, J., Ibrahim, M., Kumar, R.: Quantification of secrecy in partially observed stochastic discrete event systems. IEEE Trans. Autom. Sci. Eng. (accepted (Sept. 2015))
4. Chen, J., Kumar, R.: Failure detection framework for stochastic discrete event systems with guaranteed error bounds. IEEE Trans. Autom. Control **60**(6), 1542–1553 (2015)
5. Christian, S., Collberg, C.T.: Watermarking, tamper-proofing, and obfuscation-tools for software protection. IEEE Trans. Softw. Eng. **28**(8), 735–746 (2002)
6. Cover, T.M., Thomas, J.A.: Elements of information theory. Wiley, New York (2012)
7. Daemen, J., Rijmen, V.: Aes proposal: Rijndael, version 2, aes submission (1999)
8. Espinoza, B., Smith, G.: Min-entropy as a resource. Inf. Comput. **226**, 57–75 (2013)
9. Garg, S., Gentry, C., Halevi, S., Raykova, M., Sahai, A., Waters, B.: Candidate indistinguishability obfuscation and functional encryption for all circuits. In: IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS' 2013), pp. 40–49, Berkeley, CA, 2013
10. Garg, V.K., Kumar, R., Marcus, S.I.: A probabilistic language formalism for stochastic discrete-event systems **44**(2), 280–293 (1999)

11. Ibrahim, M., Chen, J., Kumar, R.: Secrecy in stochastic discrete event systems. In: Proceedings of 11th IEEE International Conference on Networking, Sensing and Control (ICNSC'14), pp. 48–53. Miami, FL, 2014
12. Ibrahim, M., Chen, J., Kumar, R.: An information theoretic measure for secrecy loss in stochastic discrete event systems. In: Proceedings of the 7th International Conference on Electronics, Computers and Artificial Intelligence (ECAI'15), pp. 1–6, Bucharest, 2015
13. Jacob, R., Lesage, J.J., Faure, J.M.: Opacity of discrete event systems: models, validation and quantification. In: Proceedings of the 5th International Workshop on Dependable Control of Discrete Systems (DCDS'15), hal-01139890, Cancun, Mexico, 2015
14. Kaijser, T.: A limit theorem for partially observed markov chains. Ann. Prob. **3**(4), 677–696 (1975)
15. Kak, A.: Aes: Lecture notes in computer and network security (2015). Purdue University, https://engineering.purdue.edu/kak/compsec/NewLectures/Lecture8.pdf. Accessed 1 May 2015
16. Kundur, D., Ahsan, K.: Practical internet steganography: Data hiding in ip. In: Proceedings of Texas Workshop on Security of Information Systems, College Station, Texas, 2003
17. Qiu, W., Kumar, R.: Distributed diagnosis under bounded-delay communication of immediately forwarded local observations. IEEE Trans. Syst. Man Cybern. Part A: Syst. Humans (2008)
18. Ren, J., Wu, J.: Survey on anonymous communications in computer networks. Comput. Commun. **33**, 420–431 (2010)
19. Saboori, A., Hadjicostis, C.N.: Opacity verification in stochastic discrete event systems. In: Proceedings of 49th IEEE Conference on Decision and Control, pp. 6759–6764, Atlanta, GA, 2010
20. Saboori, A., Hadjicostis, C.N.: Probabilistic current-state opacity is undecidable. In: Proceedings of 19th International Symposium on Mathematical Theory Network and Systems (MTNS '2010), pp. 477–483, Budapest, Hungary, 2010
21. Smith, G.: On the foundations of quantitative information flow. In: Proceedings of International Conference on Foundations of Software Science and Computation Structures (FoSSaCS 09), pp. 288–302, 2009
22. Takai, S., Kumar, R.: Verification and synthesis for secrecy in discrete-event systems. In: Proceedings of IEEE American Control Conference, (ACC '09), pp. 4741–4746, St. Louis, MO, 2009
23. Wang, X., Ray, A.: A language measure for performance evaluation of discrete-event supervisory control systems. Appl. Math. Model. **28**(9), 817–833 (2004)
24. Xie, A., Beerel, P.A.: Efficient state classification of finite-state Markov chains. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **17**(12), 1334–1339 (1998)
25. Zhang, T., Lee, R.B.: Secure cache modeling for measuring side-channel leakage. Technical Report, Princeton University (2014)