Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Verification of Infinite-step Opacity Using Labeled Petri Nets

Hao Lan [*,**] **Yin Tong** [*] **Carla Seatzu** [**]

[*] *School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, China (e-mail: haolan@my.swjtu.edu.cn, yintong@swjtu.edu.cn(Corresponding Author)).*
[**] *Department of Electrical and Electronic Engineering, University of Cagliari, 09123 Cagliari, Italy (e-mail: seatzu@diee.unica.it)*

Abstract: Opacity is an important information secure property. A system is said to be infinite-step opaque if the intruder is never able to ascertain that the system is or has been in a secret state at some time, based on its observation of the system evolution. This work aims to verify infinite-step opacity of discrete event systems modeled with labeled Petri nets. Based on the notion of basis reachability graph, a new structure called *basis two-way observer* is proposed to check infinite-step opacity of a bounded system, which is shown to be more efficient than the standard method based on the reachability graph.

Keywords: Discrete event systems, Petri nets, Infinite-step opacity.

1. INTRODUCTION

Motivated by the concern about security and privacy in cyber-physical systems, opacity has been extensively investigated in the past years. Opacity describes the ability of a system to hide a secret behavior from the intruders. Different notions of opacity properties have been defined for discrete event systems (DESs), including language-based opacity (Lin, 2011; Zhang et al., 2012), current-state opacity (Wu and Lafortune, 2013; Cong et al., 2018), initial-state opacity (Saboori and Hadjicostis, 2013; Tong et al., 2015b), K-step opacity (Falcone and Marchand, 2015; Yin et al., 2019), infinite-step opacity (Saboori and Hadjicostis, 2011; Yin and Lafortune, 2017), etc. In this paper, we focus on *infinite-step opacity*. Given a set of secret states, a system is said to be *infinite-step opaque* if the intruder is not able (and will never be able) to infer if the system is in a secret state, or if it has been in a secret state at some time instant.

The notion of infinite-step opacity was first defined by Saboori and Hadjicostis (2009) in the nondeterministic finite automaton (NFA) framework assuming that the events are partially observable. Later on, the opacity property was deeply explored in Saboori and Hadjicostis (2011), where it is shown that infinite-step opacity can be verified by constructing a current-state estimator and a bank of estimators for a given NFA, and the verification of infinite-step opacity is proved to be PSPACE-hard. More efficient approaches to check infinite-step opacity are proposed by Yin and Lafortune (2017). Such approaches are based on the construction of a new structure, called *two-way observer* (TW-observer). It is built through the concurrent composition of two observers: one is the observer of the given automaton, and the other is the observer of its reversed automaton. Yin and Lafortune (2017) show that the complexity of verifying infinite-step opacity is exponential in the number of states of the system.

Petri nets have been extensively used to model and check different types of opacity, e.g., initial-state opacity (Tong et al., 2015b), current-state opacity (Tong et al., 2015a; Cong et al., 2018), and language-based opacity (Tong et al., 2016). Moreover, these problems can be more effectively solved in the framework of Petri nets for their structural analysis and algebraic techniques. However, to the best of our knowledge, infinite-step opacity has never been studied in the framework of Petri nets.

In this paper, we address the formalization and verification of infinite-step opacity in bounded labeled Petri net systems. The secret is defined as a subset of the reachable markings. A labeled Petri net (LPN) system is said to be infinite-step opaque with respect to a given secret if the intruder can never infer that the system is or it has been in a secret state.

The *basis reachability graph* (BRG) of an LPN system summarizes in a compact form the information contained in its *reachability graph* (RG) [1]. Each node in the BRG represents not only the marking associated with it, but also its unobservable reach. In addition, only markings (called *basis markings*) reachable through observable transitions and the unobservable transition sequences whose firing is necessary to enable the observable transitions, are enumerated. As a consequence, the size of the BRG is usually much smaller than that of the RG, thus the BRG has been efficiently used to verify some opacity properties (Tong et al., 2017). In this paper, a necessary and sufficient condition for infinite-step opacity is presented, which relies on the language of the RG. We also show that such a condition can be reduced to a condition on the language of the BRG under certain assumptions. More precisely, if any basis marking is in the secret, the markings that can be reached from it through unobservable transitions are

[1] The BRG and RG are both automata.

also contained in the secret, then the BRG can be used to verify infinite-step opacity. A new structure called the *basis two-way observer* (BTW-observer) is constructed in order to characterize the relation among the languages generated in the BRG. Note that the BTW-observer is not the TW-observer of the BRG. As illustrated in Algorithm 1, the construction of the BTW-observer follows different rules with respect to the TW-observer and it results in an automaton with a lower number of transitions.

The contributions of the paper can be summarized as follows.

- Infinite-step opacity is formally defined in the framework of labeled Petri nets.
- A necessary and sufficient condition for infinite-step opacity based on the RG of the Petri net system is presented.
- Under certain assumptions, such a condition can be rewritten in terms of a new condition on the BRG. Therefore, the enumeration of all the reachable markings is avoided, thus allowing to deal with much larger systems.
- A new structure, the BTW-observer of the BRG, is proposed to verify infinite-step opacity.

Note that, due to space constraints, we refer the reader to Lan et al. (2020) for full proofs of the results in the paper. The rest of the paper is organized as follows. In Section 2 some background on finite automata and labeled Petri nets is provided. The definition of infinite-step opacity in labeled Petri net systems is formalized in Section 3. In Section 4, a necessary and sufficient condition for infinite-step opacity is provided. The construction of the BTW-observer is presented in Section 5. Conclusions are finally drawn in Section 6 where our future lines of research in this framework are also illustrated.

2. PRELIMINARIES AND BACKGROUND

In this section we recall the formalisms used in the paper and some results on state estimation in labeled Petri nets. For more details, we refer the reader to (Murata, 1989; Cassandras and Lafortune, 2008).

2.1 Automata

A *nondeterministic finite (state) automaton* (NFA) is a 4-tuple $A = (X, E, f, X_0)$, where X is the finite *set of states*, E is the finite *set of events*, $f : X \times E \to 2^X$ is the (partial) *transition relation* [2], and $X_0 \subseteq X$ is the set of *initial states*. The transition relation f can be extended to $f : X \times E^* \to 2^X$ in a standard manner. Given an initial state $x_0 \in X_0$ and an event sequence $w \in E^*$, $f(x_0, w) \neq \emptyset$ is the set of states reached in A from x_0 with w occurring and it is denoted by $f(x_0, w)!$. The *reversed automaton* $A_r = (X, E, f_r, X)$ of A is the automaton obtained by reversing all arcs in A and taking all the states in A as its initial states.

Given a set of states $Y \subseteq X$, the language generated from Y is $\mathcal{L}(A, Y) = \{w \in E^* | \exists x \in Y : f(x, w)!\}$. If $Y = \{x\}$ is a singleton, the generated language is simply denoted by $\mathcal{L}(A, x)$. The language generated from the initial states is denoted by $\mathcal{L}(A)$.

Given an NFA, its equivalent DFA (namely the DFA generating the same language) called *observer* can be constructed following the procedure in Section 2.3.4 of (Cassandras and Lafortune, 2008). Each state of the observer is a subset of states of X in which the NFA may be after a certain event sequence has occurred.

2.2 Petri Nets

A *Petri net* is a structure $N = (P, T, Pre, Post)$, where P is a set of m *places*, graphically represented by circles; T is a set of n *transitions*, graphically represented by bars; $Pre : P \times T \to \mathbb{N}$ and $Post : P \times T \to \mathbb{N}$ are the *pre-* and *post-incidence functions* that specify the arcs directed from places to transitions, and from transitions to places, respectively. The *incidence matrix* of a net is denoted by $C = Post - Pre$. A Petri net is *acyclic* if there are no oriented cycles.

A *marking* is a vector $M : P \to \mathbb{N}$ that assigns to each place a non-negative integer number of tokens, graphically represented by black dots. The marking of place p is denoted by $M(p)$. A marking is also denoted by $M = \sum_{p \in P} M(p) \cdot p$. A *Petri net system* $\langle N, M_0 \rangle$ is a net N with *initial marking* M_0. A transition t is *enabled* at marking M if $M \geq Pre(\cdot, t)$ and may fire yielding a new marking $M' = M + C(\cdot, t)$. We write $M[\sigma\rangle$ to denote that the sequence of transitions $\sigma = t_{j1} \cdots t_{jk}$ is enabled at M, and $M[\sigma\rangle M'$ to denote that the firing of σ yields M'. The set of all enabled transition sequences in N from marking M is $L(N, M) = \{\sigma \in T^* | M[\sigma\rangle\}$. Given a transition sequence $\sigma \in T^*$, the function $\pi : T^* \to \mathbb{N}^n$ associates with σ the Parikh vector $y = \pi(\sigma) \in \mathbb{N}^n$, where $y(t) = k$ if transition t appears k times in σ. Given a sequence of transitions $\sigma \in T^*$, its *prefix* (denoted by $\sigma' \preceq \sigma$) is a string for which $\exists \sigma'' \in T^* : \sigma' \sigma'' = \sigma$. The *length* of σ is denoted by $|\sigma|$.

A marking M is *reachable* in $\langle N, M_0 \rangle$ if there exists a transition sequence σ such that $M_0[\sigma\rangle M$. The set of all markings reachable from M_0 defines the *reachability set* $R(N, M_0)$ of $\langle N, M_0 \rangle$. A Petri net system is *bounded* if there exists a non-negative integer $k \in \mathbb{N}$ such that for any place $p \in P$ and any reachable marking $M \in R(N, M_0)$, $M(p) \leq k$ holds.

A *labeled Petri net* (LPN) system is a 4-tuple $G = (N, M_0, E, \ell)$, where $\langle N, M_0 \rangle$ is a Petri net system, E is the *alphabet* (a set of labels) and $\ell : T \to E \cup \{\varepsilon\}$ is the *labeling function* that assigns to each transition $t \in T$ either a symbol from E or the empty word ε. Therefore, the set of transitions can be partitioned into two disjoint sets $T = T_o \cup T_u$, where $T_o = \{t \in T | \ell(t) \in E\}$ is the set of observable transitions and $T_u = T \setminus T_o = \{t \in T | \ell(t) = \varepsilon\}$ is the set of unobservable transitions. We denote $n_o = |T_o|$ (resp. $n_u = |T_u|$) the number of observable (resp. unobservable) transitions. Given a marking $M \in R(N, M_0)$, we define

$$U(M) = \{M' \in \mathbb{N}^m | M[\sigma_u\rangle M', \sigma_u \in T_u^*\}$$

its *unobservable reach*, namely, the set of markings reachable from M through unobservable transitions. Given a subset of markings $Y \subseteq R(N, M_0)$, $U(Y) = \bigcup_{M \in Y} U(M)$.

[2] In the paper we do not use the standard NFA whose transition relation is defined on $X \times (E \cup \{\varepsilon\})$. Namely, we assume that no transition is labeled with the empty word.

The labeling function can be extended to transition sequences $\ell : T^* \to E^*$ as $\ell(\sigma t) = \ell(\sigma)\ell(t)$ with $\sigma \in T^*$ and $t \in T$. Given a set $Y \subseteq R(N, M_0)$ of markings, the *language generated by G from Y* is

$$\mathcal{L}(G, Y) = \bigcup_{M \in Y} \{w \in E^* | \exists \sigma \in L(N, M) : w = \ell(\sigma)\}.$$

In particular, the *language generated by G* is

$$\mathcal{L}(G, \{M_0\}) = \{w \in E^* | \exists \sigma \in L(N, M_0) : w = \ell(\sigma)\},$$

which is also simply denoted by $\mathcal{L}(G)$. It is the set of words that can be observed by the intruder. A word $w \in \mathcal{L}(G)$ is called an *observation*. We denote as

$$\mathcal{C}(w) = \{M \in \mathbb{N}^m | \exists \sigma \in L(N, M_0) : M_0[\sigma\rangle M, \ell(\sigma) = w\}$$

the set of markings *consistent* with w.

Given an LPN system $G = (N, M_0, E, \ell)$, the *T_u-induced subnet* $N' = (P, T', Pre', Post')$ of N, is the net that results by removing all transitions in T_o from N, where Pre' and $Post'$ are the restrictions of Pre, $Post$ to T_u, respectively. The incidence matrix of the T_u-induced subnet is denoted by $C_u = Post' - Pre'$.

3. INFINITE-STEP OPACITY IN LABELED PETRI NETS

Infinite-step opacity has been defined in the automaton framework (Saboori and Hadjicostis, 2011; Yin and Lafortune, 2017). In this section we extend this notion to labeled Petri net systems.

In the framework of LPNs, the *secret* is a subset of reachable makings $S \subseteq R(N, M_0)$. A marking $M \in S$ is called a *secret marking*. Markings in $\bar{S} = R(N, M_0) \setminus S$ are called *exposable markings*.

Definition 1. [**Infinite-Step Opacity**] Let $G = (N, M_0, E, \ell)$ be an LPN system and $S \subseteq R(N, M_0)$ a secret. System G is *infinite-step opaque* w.r.t S if $\forall \sigma_1\sigma_2 \in L(N, M_0)$ with $M_0[\sigma_1\rangle M_1 \in S$, there exists $\sigma_1'\sigma_2' \in L(N, M_0)$ such that $M_0[\sigma_1'\rangle M_1' \notin S$, $\ell(\sigma_1) = \ell(\sigma_1')$, and $\ell(\sigma_2) = \ell(\sigma_2')$.

In words, an LPN system is infinite-step opaque if for any transition sequence σ_1 leading to a secret marking M_1 there exists another transition sequence σ_1' that leads to an exposable marking M_1' producing the same observation. Meanwhile, the same observations can be generated from both M_1 and M_1'. Namely, for any observation $\ell(\sigma_1\sigma_2) \in \mathcal{L}(G)$ the intruder never knows that the system once visited a secret marking.

Example 2. Let us consider the LPN system in Fig. 1(a) where $T_o = \{t_2, t_3, t_6, t_7, t_8, t_9\}$ and $T_u = \{t_1, t_4, t_5\}$. The RG of the LPN system is shown in Fig. 1(b). Let the secret be $S = \{M_2, M_4\}$. Let $\sigma_1 = t_1t_2t_4$ and $\sigma_2 = t_6t_8$. Thus, $M_0[\sigma_1\rangle M_4 \in S$. However, there is no $\sigma_1'\sigma_2' \in L(N, M_0)$ such that $M_0[\sigma_1'\rangle M \notin S$ and $\ell(\sigma_1'\sigma_2') = \ell(\sigma_1\sigma_2) = aaa$. Therefore, the LPN system is not infinite-step opaque.
Now, suppose that transition t_9 is labeled a instead of b. The LPN system becomes infinite-step opaque w.r.t S since the intruder will never be able to distinguish markings M_2 and M_4 from M_3 and M_5, thus, the intruder is unable to ascertain if the secret markings have been visited when observing aa^*.

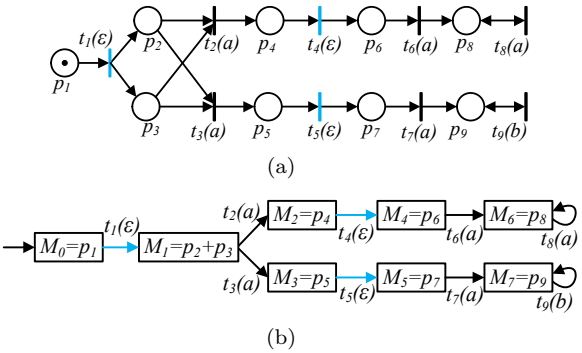

Fig. 1. The LPN system in Example 2 (a), and its RG (b).

Given a secret S and an observation $w \in \mathcal{L}(G)$, we denote as $\mathcal{S}(w) = \mathcal{C}(w) \cap S$ (resp. $\overline{\mathcal{S}}(w) = \mathcal{C}(w) \setminus S$) the set of secret (resp. exposable) markings consistent with the observation.

Proposition 3. Let $G = (N, M_0, E, \ell)$ be an LPN system and $S \subseteq R(N, M_0)$ a secret. System G is infinite-step opaque w.r.t S if and only if $\forall w \in \mathcal{L}(G)$, $\mathcal{L}(G, \mathcal{S}(w)) \subseteq \mathcal{L}(G, \overline{\mathcal{S}}(w))$.

Proof. Follows from Definition 1. □

According to the above proposition, an LPN system is infinite-step opaque w.r.t a given secret if and only if for any observation $w \in \mathcal{L}(G)$ the language generated from its consistent secret markings $\mathcal{S}(w)$ is contained in the language generated from its consistent exposable markings. Therefore, the problem of verifying infinite-step opacity in LPN systems is transformed into a language containment problem in G.

4. NECESSARY AND SUFFICIENT CONDITIONS FOR INFINITE-STEP OPACITY

In this section, we prove that, under an appropriate assumption, a language containment problem in the RG can be reduced to a language containment problem in the *basis reachability graph* (BRG), which is a compact representation of the RG. Thus exhaustively enumerating all states in the RG is avoided. We first recall the definition of basis marking and the construction rules of the BRG presented in Ma et al. (2017).

Definition 4. Given a marking M and an observable transition $t \in T_o$, we denote as

$$\Sigma(M, t) = \{\sigma \in T_u^* | M[\sigma\rangle M', M' \geq Pre(\cdot, t)\}$$

the set of *explanations* of t at M and

$$Y(M, t) = \{y_u \in \mathbb{N}^{n_u} | \exists \sigma \in \Sigma(M, t) : y_u = \pi(\sigma)\}$$

the set of *e-vectors*.

After firing any unobservable transition sequence in $\Sigma(M, t)$ at M, transition t is enabled. To provide a compact representation of the reachability set, we are interested in finding the explanations whose firing vector is minimal.

Definition 5. Given a marking M and an observable transition $t \in T_o$, we denote as

$$\Sigma_{min}(M, t) = \{\sigma \in \Sigma(M, t) | \nexists \sigma' \in \Sigma(M, t) : \pi(\sigma') \lneqq \pi(\sigma)\}$$

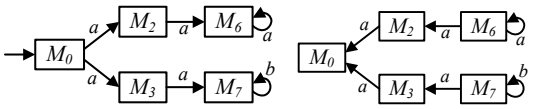

Fig. 2. The BRG $\mathcal{B}$(a), and the reversed BRG $\mathcal{B}_r$, where all the states are initial states (b).

the set of *minimal explanations* of t at M and

$$Y_{min}(M,t) = \{y_u \in \mathbb{N}^{n_u} | \exists \sigma \in \Sigma_{min}(M,t) : y_u = \pi(\sigma)\}$$

the corresponding set of *minimal e-vectors*.

There are many approaches to calculate $Y_{min}(M,t)$. In particular, Cabasino et al. (2011) present an approach that only requires algebraic manipulations when the T_u-induced subnet is acyclic.

Definition 6. Given an LPN system $G = (N, M_0, E, \ell)$ whose T_u-induced subnet is acyclic, its *basis marking set* $\mathcal{M}_b$ is defined as follows:

- $M_0 \in \mathcal{M}_b$;
- if $M \in \mathcal{M}_b$, then $\forall t \in T_o, y_u \in Y_{min}(M,t)$,
$$M' = M + C(\cdot,t) + C_u \cdot y_u \Rightarrow M' \in \mathcal{M}_b.$$

A marking $M_b \in \mathcal{M}_b$ is called a *basis marking* of G.

The set of basis markings contains the initial marking and all other markings that are reachable from a basis marking by firing a transition sequence $\sigma_u t$, where $t \in T_o$ and $\sigma_u \in \Sigma_{min}(M,t)$. By Definition 6, basis markings can be recursively computed from the initial marking if the T_u-induced subnet is acyclic. Note that since $y_u \in Y_{min}(M,t)$, t is enabled at some marking in the unobservable reach of M. Clearly, $\mathcal{M}_b \subseteq R(N, M_0)$ and in practical cases the number of basis markings is much smaller than the number of reachable markings (Tong et al., 2017; Cabasino et al., 2011; Ma et al., 2017). The number of basis markings is finite if the LPN system is bounded. Therefore, hereafter it is assumed that:

A1) the LPN system is bounded, and
A2) its T_u-induced subnet is acyclic.

Given an LPN system $G = (N, M_0, E, \ell)$, its BRG is an NFA, where each state is a basis marking, the set of events is the alphabet of the LPN system, and there is no transition labeled with the empty word. We denote it as $\mathcal{B} = (\mathcal{M}_b, E, f, M_0)$. Due to limited spaces here, the detailed algorithm for constructing the BRG is not presented and Tong et al. (2017) is referred. Clearly, $\mathcal{L}(\mathcal{B}) = \mathcal{L}(G)$.

Example 7. Let us consider again the LPN system in Fig. 1(a). The system satisfies Assumptions A1 and A2, and its BRG is shown in Fig. 2(a), where there are only five basis markings $\mathcal{M}_b = \{M_0, M_2, M_3, M_6, M_7\}$.

Given an LPN system G, its set of basis markings $\mathcal{M}_b$, and an observation $w \in \mathcal{L}(G)$, in Ma et al. (2017) it is shown that

$$\mathcal{C}(w) = U(\mathcal{M}_b \cap \mathcal{C}(w)). \tag{1}$$

Given an observation $w \in \mathcal{L}(G)$, the set of *secret basis markings consistent with w is denoted by* $\mathcal{S}_b(w) = \mathcal{S}(w) \cap \mathcal{M}_b$, *and the set of* exposable *basis markings consistent with* w is denoted by $\overline{\mathcal{S}}_b(w) = \overline{\mathcal{S}}(w) \cap \mathcal{M}_b$. Since $\mathcal{L}(\mathcal{B}, M_b) = \mathcal{L}(G, U(M_b)) = \mathcal{L}(G, M_b)$, $\mathcal{L}(\mathcal{B}, \mathcal{S}_b(w)) = \mathcal{L}(G, U(\mathcal{S}_b(w))) = \mathcal{L}(G, \mathcal{S}_b(w))$. Then, since $\mathcal{S}_b(w) \subseteq \mathcal{S}(w)$,

$$\mathcal{L}(\mathcal{B}, \mathcal{S}_b(w)) \subseteq \mathcal{L}(G, \mathcal{S}(w)). \tag{2}$$

Analogously, $\mathcal{L}(\mathcal{B}, \overline{\mathcal{S}}_b(w)) = \mathcal{L}(G, \overline{\mathcal{S}}_b(w))$ and

$$\mathcal{L}(\mathcal{B}, \overline{\mathcal{S}}_b(w)) \subseteq \mathcal{L}(G, \overline{\mathcal{S}}(w)). \tag{3}$$

Thanks to the above inclusion relationships, the following proposition can be derived.

Proposition 8. Let G be an LPN system, $\mathcal{B}$ its BRG, and $w \in \mathcal{L}(G)$ an observation. If $\mathcal{L}(\mathcal{B}, \mathcal{S}_b(w)) \subseteq \mathcal{L}(\mathcal{B}, \overline{\mathcal{S}}_b(w))$, then $\mathcal{L}(G, \mathcal{S}(w)) \subseteq \mathcal{L}(G, \overline{\mathcal{S}}(w))$.

Theorem 9. Let G be an LPN system and $S \subseteq R(N, M_0)$ a secret. System G is infinite-step opaque w.r.t S if $\mathcal{L}(\mathcal{B}, \mathcal{S}_b(w)) \subseteq \mathcal{L}(\mathcal{B}, \overline{\mathcal{S}}_b(w))$.

Theorem 9 provides a sufficient condition for infinite-step opacity. Furthermore, the condition only relies on the structure of the BRG. Next we show that a necessary and sufficient condition in the BRG can be derived under the additional assumption:

A3) $M_b \in S \Rightarrow U(M_b) \subseteq S$.

Namely, if a basis marking M_b is a secret marking, then any marking reachable from M_b firing unobservable transitions is also a secret marking.

Lemma 10. Let G be an LPN system and S a secret satisfying Assumption A3. It holds that $\mathcal{L}(\mathcal{B}, \overline{\mathcal{S}}_b(w)) = \mathcal{L}(G, \overline{\mathcal{S}}(w))$, for any $w \in \mathcal{L}(G)$.

Based on Lemma 10, the condition in Proposition 8 becomes necessary and sufficient under the additional Assumption A3.

Proposition 11. Let G be an LPN system, S a secret, which satisfy Assumptions A1 to A3. It holds that $\mathcal{L}(G, \mathcal{S}(w)) \subseteq \mathcal{L}(G, \overline{\mathcal{S}}(w))$ if and only if $\mathcal{L}(\mathcal{B}, \mathcal{S}_b(w)) \subseteq \mathcal{L}(\mathcal{B}, \overline{\mathcal{S}}_b(w))$.

In other words, the language containment problem in the RG is reduced to a language containment problem in the BRG. Thus, a necessary and sufficient condition for infinite-step opacity based on the BRG is obtained.

Theorem 12. An LPN System G is infinite-step opaque w.r.t a given secret S under Assumptions A1 to A3, if and only if

$$\forall w \in \mathcal{L}(\mathcal{B}), \quad \mathcal{L}(\mathcal{B}, \mathcal{S}_b(w)) \subseteq \mathcal{L}(\mathcal{B}, \overline{\mathcal{S}}_b(w)), \tag{4}$$

where $\mathcal{B}$ is the BRG of G.

Theorem 12 leads to the conclusion that the BRG can be used to check infinite-step opacity of an LPN system. In the next section, we propose a new structure called *basis two-way observer* (BTW-observer), to verify the condition in Eq. (4).

5. BASIS TWO-WAY OBSERVER

A structure called two-way observer (TW-observer) is proposed to verify infinite-step opacity in the automaton framework (Yin and Lafortune, 2017). Its construction is based on the synchronization between the observer and the initial-state estimator [3] of the system. The BTW-observer

[3] The initial-state estimator of an NFA is the observer of its reversed NFA with the set of initial-states being the whole state space (Wu and Lafortune, 2013).

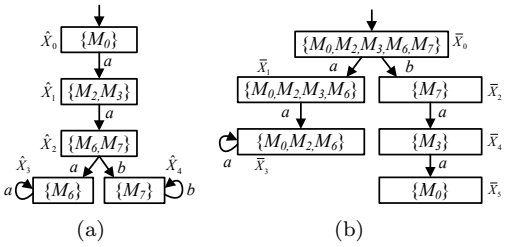

Fig. 3. The observer $\mathcal{B}_o$ (a) and the initial-state estimator $\mathcal{B}_e$ (b) of the BRG in Fig. 2 (a).

we propose, is different from the TW-observer as it is constructed through partial synchronization between the observer and the initial-state estimator of the BRG. We first introduce some notations that are helpful to formalize the construction of the BTW-observer.

Given a BRG $\mathcal{B} = (\mathcal{M}_b, E, f, M_0)$, we associate with it the following three automata:

- $\mathcal{B}_r = (\mathcal{M}_b, E, f_r, \mathcal{M}_b)$ is the reversed automaton of $\mathcal{B}$, which is obtained by reversing all arcs in $\mathcal{B}$ and taking all the states in $\mathcal{B}$ as initial states.
- $\mathcal{B}_o = (\mathcal{X}, E, f_o, \hat{X}_0)$ is the observer of $\mathcal{B}$, where $\mathcal{X} \subseteq 2^X$ is a finite set of states, $\hat{X}_0 = \{M_0\}$ is the initial state. The event set of $\mathcal{B}_o$ is the alphabet E. The transition function is $f_o : \mathcal{X} \times E \to \mathcal{X}$. The observer of the BRG can be constructed by applying the standard determination algorithm in Cassandras and Lafortune (2008).
- $\mathcal{B}_e = (\mathcal{X}_e, E, f_e, \overline{X}_0)$ is the initial-state estimator of $\mathcal{B}$. $\mathcal{B}_e$ is also the observer of the reversed automaton of $\mathcal{B}$, with the initial state $\overline{X}_0 = \mathcal{M}_b$.

Note that $\mathcal{L}(\mathcal{B}_o) = \mathcal{L}(\mathcal{B})$. Therefore, given $w \in \mathcal{L}(\mathcal{B})$,

$$f_o(\hat{X}_0, w) = \mathcal{C}(w) \cap \mathcal{M}_b. \tag{5}$$

Let $w \in \mathcal{L}(\mathcal{B}_e)$ be an observation, and w^r the reversed word of w. It holds that

$$f_e(\overline{X}_0, w) = \{M \in \mathcal{M}_b | f(M, w^r)!\}. \tag{6}$$

Namely, $f_e(\overline{X}_0, w)$ is the set of basis markings from which the reversed word of w can be generated in $\mathcal{B}$, and consequently, in the system G.

Let $w_1 \in \mathcal{L}(\mathcal{B})$ and $w_2^r \in \mathcal{L}(\mathcal{B}_e)$. We denote

$$\mathcal{M}(w_1|w_2) = f_o(\hat{X}_0, w_1) \cap f_e(\overline{X}_0, w_2^r). \tag{7}$$

By Eqs. (5) and (6), if $\mathcal{M}(w_1|w_2) \neq \emptyset$, $\mathcal{M}(w_1|w_2)$ is the set of basis markings consistent with w_1 and from which w_2 can be generated in the BRG, and thus in the system G.

Theorem 13. Let $\mathcal{B}$ be the BRG of an LPN system G, $\mathcal{B}_o$ the observer of $\mathcal{B}$, $\mathcal{B}_e$ the initial-state estimator of $\mathcal{B}$, and S a secret. System G is not infinite-step opaque w.r.t S if and only if $\exists w_1 \in \mathcal{L}(\mathcal{B}), w_2^r \in \mathcal{L}(\mathcal{B}_e)$ such that

$$\mathcal{M}(w_1|w_2) \subseteq S \text{ and } \mathcal{M}(w_1|w_2) \neq \emptyset. \tag{8}$$

In the following, the BTW-observer is proposed in order to check whether the conditions in Theorem 13 are satisfied.

Let $\mathcal{B}_o = (\mathcal{X}, E, f_o, \hat{X}_0)$ be the observer of the BRG of an LPN system G, and $\mathcal{B}_e = (\mathcal{X}_e, E, f_e, \overline{X}_0)$ be its initial-state estimator. The *BTW-observer* of G is a DFA $\mathcal{B}_{tw} = (Q, E_{tw}, f_{tw}, q_0)$, where $Q \subseteq \mathcal{X} \times \mathcal{X}_e$ is a finite set of states, $E_{tw} = (E \times \{\lambda\}) \cup (\{\lambda\} \times E)$ is the set of events, $f_{tw} : Q \times E_{tw} \to Q$ is the transition function, and $q_0 = (\hat{X}_0, \overline{X}_0)$ is the initial state. Each state q in $\mathcal{B}_{tw}$ consists of two elements $(\hat{X}_i, \overline{X}_j)$, and we denote $q(1) = \hat{X}_i$ its first element and $q(2) = \overline{X}_j$ its second element.

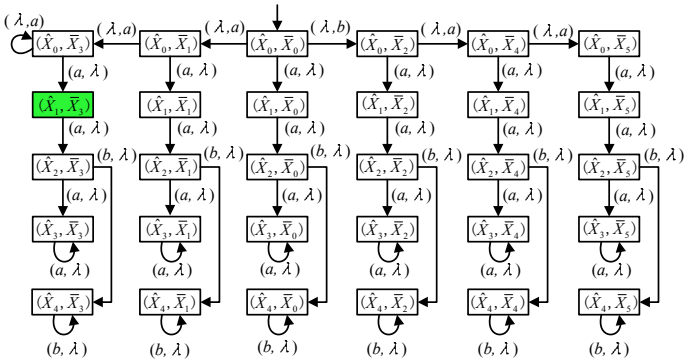

Fig. 4. The BTW-observer of the LPN system in Fig. 1(a).

The procedure to construct the BTW-observer is summarized in Algorithm 1, whose main steps can be explained as follows. The initial state of the BTW-observer is taken equal to $q_0 = (\hat{X}_0, \overline{X}_0)$. Two sets are initialized at q_o (Step 1): set Q, which at the end of the algorithm contains all the states of the BTW-observer; set Q_{new}, which contains all the states of the BTW-observer that still need to be explored. The algorithm terminates when Q_{new} is equal to the empty set. Steps 2 to 10 define all the states q' reachable through strings in $(\lambda, E)^*$. Then, from all the states obtained so far, we compute the states q' reachable through $(E, \lambda)^*$ (Steps 11 to 20).

Example 14. Consider again the LPN system in Fig. 1(a). Its BTW-observer is shown in Fig. 4. For instance, let $\sigma = (\lambda, b)(a, \lambda)(a, \lambda)$. Then $w_1 = aa$, $w_2 = b$, and $f_{tw}(q_0, \sigma) = (\hat{X}_2, \overline{X}_2)$ with $\hat{X}_2 = \{M_6, M_7\}$ and $\overline{X}_2 = \{M_7\}$. Indeed, $f_o(\hat{X}_0, w_1) = \hat{X}_2$ and $f_e(\overline{X}_0, b) = \overline{X}_2$, and $\mathcal{M}(w_1|w_2^r) = \hat{X}_2 \cap \overline{X}_2 = \{M_7\}$.

The following theorem shows how the BTW-observer can be used to verify infinite-step opacity of an LPN system.

Theorem 15. Let G be an LPN system, S a secret, and $\mathcal{B}_{tw} = (Q, E_{tw}, f_{tw}, q_0)$ the BTW-observer. System G is infinite-step opaque w.r.t S if and only if $\forall q = (q(1), q(2)) \in Q$,

$$q(1) \cap q(2) \nsubseteq S \ \vee \ q(1) \cap q(2) = \emptyset.$$

Example 16. Consider again the LPN system in Fig. 1(a), where the secret is $S = \{M_2, M_4\}$. Its reversed automaton, observer and initial-state estimator are shown in Figs. 2(b), 3(a), and 3(b), respectively. By Algorithm 1, the BTW-observer of the LPN system is constructed as reported in Fig. 4. Since there exists a state $(\hat{X}_1, \overline{X}_3)$ with $\hat{X}_1 \cap \overline{X}_3 = \{M_2\} \subseteq S$, by Theorem 15, the LPN system is not infinite-step opaque w.r.t S.

We discuss the computational complexity of Algorithm 1. In the BTW-observer, in the worst case, there are $2^{|\mathcal{M}_b|} \times 2^{|\mathcal{M}_b|}$ states and $|E| \times 2^{|\mathcal{M}_b|} \times 2^{|\mathcal{M}_b|} + |E| \times 2^{|\mathcal{M}_b|}$ transitions. Therefore, the complexity of the proposed algorithm is $\mathcal{O}(|E| \times 2^{|\mathcal{M}_b|} \times 2^{|\mathcal{M}_b|})$. Since $|\mathcal{M}_b| << |R(N, M_0)|$, the proposed approach is more efficient than the one using

Algorithm 1 Computation of the BTW-observer

Input: An observer $\mathcal{B}_o = (\mathcal{X}, E, f_o, \hat{X}_0)$; An initial-state estimator $\mathcal{B}_e = (\mathcal{X}_e, E, f_e, \overline{X}_0)$.
Output: The BTW-observer $\mathcal{B}_{tw} = (Q, E_{tw}, f_{tw}, q_0)$.

$q_0 := (\hat{X}_0, \overline{X}_0)$, $Q := \{q_0\}$, $Q_{new} := \{q_0\}$.
for all $q = (q(1), q(2)) \in Q_{new}$, **do**
  for all $e \in E$: $f_e(q(2), e)!$, **do**
    $q' := (q(1), f_e(q(2), e))$, $f_{tw}(q, (\lambda, e)) := q'$,
    if $q' \notin Q$, **then**
      $Q := Q \cup \{q'\}$, $Q_{new} := Q_{new} \cup \{q'\}$,
    end if
  end for
  $Q_{new} := Q_{new} \setminus \{q\}$.
end for
$Q_{new} := Q$.
for all $q = (q(1), q(2)) \in Q_{new}$, **do**
  for all $e \in E$: $f_o(q(1), e)!$, **do**
    $q' := (f_o(q(1), e), q(2))$, $f_{tw}(q, (e, \lambda)) := q'$,
    if $q' \notin Q$, **then**
      $Q := Q \cup \{q'\}$, $Q_{new} := Q_{new} \cup \{q'\}$,
    end if
  end for
  $Q_{new} := Q_{new} \setminus \{q\}$.
end for

the RG. Note that given the observer and the initial-state estimator, the BTW-observer has the same set of states of the TW-observer. However, in the TW-observer in the worst case, there are $2 \times |E| \times 2^{|\mathcal{M}_b|} \times 2^{|\mathcal{M}_b|}$ transitions. For instance, the TW-observer of the observer and initial-state estimator in Fig. 3, contains the word $(a, \lambda)(\lambda, a)$ that does not appear in the BTW-observer. Moreover, all the transitions in the BTW-observer also exist in the TW-observer. Therefore, the BTW-observer is typically smaller than the TW-observer.

6. CONCLUSION

In this paper, the notion of infinite-step opacity of labeled Petri net systems is formalized. Under some assumptions on the secret and the net structure, necessary and sufficient conditions for infinite-step opacity are derived. A new structure called basis two-way observer (BTW-observer) of the basis reachability graph (BRG) is constructed to verify infinite-step opacity. The proposed approach has computational complexity advantages over the reachability graph based approaches in the literature, since enumerations of the whole state space of the system may be avoided. Our future research will focus on the verification of K-step opacity in Petri nets.

ACKNOWLEDGEMENTS

This work has been partially supported by the National Natural Science Foundation of China under Grant No. 61803317 and Grant No. 61950410604. It has also been partially supported by Project RASSR05871 MOSIMA financed by Region Sardinia, FSC 2014-2020, annuity 2017, Subject area 3, Action Line 3.1.

REFERENCES

Cabasino, M.P., Giua, A., Pocci, M., and Seatzu, C. (2011). Discrete event diagnosis using labeled Petri nets. An application to manufacturing systems. *Control Engineering Practice*, 19(9), 989–1001.

Cassandras, C.G. and Lafortune, S. (2008). *Introduction to discrete event systems*. Springer.

Cong, X., Fanti, M.P., Mangini, A.M., and Li, Z. (2018). On-line verification of current-state opacity by petri nets and integer linear programming. *Automatica*, 94, 205–213.

Falcone, Y. and Marchand, H. (2015). Enforcement and validation (at runtime) of various notions of opacity. *Discrete Event Dynamic Systems*, 25(4), 531–570.

Lan, H., Tong, Y., and Seatzu, C. (2020). Proofs of the results in "Verification of Infinite-step Opacity Using Labeled Petri Nets". doi:10.13140/RG.2.2.22842.54720.

Lin, F. (2011). Opacity of discrete event systems and its applications. *Automatica*, 47(3), 496–503.

Ma, Z., Tong, Y., Li, Z., and Giua, A. (2017). Basis marking representation of Petri net reachability spaces and its application to the reachability problem. *IEEE Transactions on Automatic Control*, 62(3), 1078–1093.

Murata, T. (1989). Petri nets: Properties, analysis and applications. *Procedings of the IEEE*, 77(4), 541–580.

Saboori, A. and Hadjicostis, C.N. (2009). Verification of infinite-step opacity and analysis of its complexity. *IFAC Proceedings Volumes*, 42(5), 46–51.

Saboori, A. and Hadjicostis, C.N. (2011). Verification of infinite-step opacity and complexity considerations. *IEEE Transactions on Automatic Control*, 57(5), 1265–1269.

Saboori, A. and Hadjicostis, C.N. (2013). Verification of initial-state opacity in security applications of discrete event systems. *Information Sciences*, 246, 115–132.

Tong, Y., Li, Z., Seatzu, C., and Giua, A. (2015a). Verification of current-state opacity using Petri nets. In *American Control Conference (ACC)*, 1935–1940. IEEE.

Tong, Y., Li, Z., Seatzu, C., and Giua, A. (2015b). Verification of initial-state opacity in Petri nets. In *54th IEEE Conference on Decision and Control (CDC)*, 344–349. IEEE.

Tong, Y., Li, Z., Seatzu, C., and Giua, A. (2017). Verification of state-based opacity using Petri nets. *IEEE Transactions on Automatic Control*, 62(6), 2823–2837.

Tong, Y., Ma, Z., Li, Z., Seactzu, C., and Giua, A. (2016). Verification of language-based opacity in Petri nets using verifier. In *American Control Conference (ACC)*, 757–763. IEEE.

Wu, Y. and Lafortune, S. (2013). Comparative analysis of related notions of opacity in centralized and coordinated architectures. *Discrete Event Dynamic Systems*, 23(3), 307–339.

Yin, X. and Lafortune, S. (2017). A new approach for the verification of infinite-step and K-step opacity using two-way observers. *Automatica*, 80, 162–171.

Yin, X., Li, Z., Wang, W., and Li, S. (2019). Infinite-step opacity and K-step opacity of stochastic discrete-event systems. *Automatica*, 99, 266–274.

Zhang, B., Shu, S., and Lin, F. (2012). Polynomial algorithms to check opacity in discrete event systems. In *24th Chinese Control and Decision Conference (CCDC)*, 763–769. IEEE.