Discrete Event Dynamic Systems (2021) 31:553–582
https://doi.org/10.1007/s10626-021-00344-2

Comparing the notions of opacity for discrete-event systems

Jiří Balun[1] · **Tomáš Masopust**[1]

Received: 13 January 2021 / Accepted: 9 June 2021 / Published online: 27 July 2021

Abstract

Opacity is an information flow property characterizing whether a system reveals its secret to a passive observer. Several notions of opacity have been introduced in the literature. We study the notions of language-based opacity, current-state opacity, initial-state opacity, initial-and-final-state opacity, K-step opacity, and infinite-step opacity. Comparing the notions is a natural question that has been investigated and summarized by Wu and Lafortune, who provided transformations among current-state opacity, initial-and-final-state opacity, and language-based opacity, and, for prefix-closed languages, also between language-based opacity and initial-state opacity. We extend these results by showing that all the discussed notions of opacity are transformable to each other. Besides a deeper insight into the differences among the notions, the transformations have applications in complexity results. In particular, the transformations are computable in polynomial time and preserve the number of observable events and determinism, and hence the computational complexities of the verification of the notions coincide. We provide a complete and improved complexity picture of the verification of the discussed notions of opacity, and improve the algorithmic complexity of deciding language-based opacity, infinite-step opacity, and K-step opacity.

Keywords Discrete event systems · Finite automata · Opacity · Transformations · Complexity

This article belongs to the Topical Collection: *Topical Collection on Cybersecurity*
Guest Editors: Rong Su and Carlos Basilio

Partially supported by the Ministry of Education, Youth and Sports under the INTER-EXCELLENCE project LTAUSA19098 and by the University projects IGA PrF 2020 019 and IGA PrF 2021 022.

✉ Jiří Balun
jiri.balun01@upol.cz

Tomáš Masopust
tomas.masopust@upol.cz

[1] Faculty of Science, Palacky University in Olomouc, Czechia, Czech Republic

1 Introduction

Applications often require to keep some information about the behavior of a system secret. Properties that guarantee such requirements include anonymity (Schneider and Sidiropoulos 1996), noninterference (Hadj-Alouane et al. 2005), secrecy (Alur et al. 2006), security (Focardi and Gorrieri 1994), and opacity (Mazarė 2004).

In this paper, we are interested in opacity for discrete-event systems (DESs) modeled by finite automata. Opacity is a state-estimation property that asks whether a system prevents an intruder from revealing the secret. The intruder is modeled as a passive observer with the complete knowledge of the structure of the system, but with only limited observation of the behavior of the system. Based on the observation, the intruder estimates the behavior of the system, and the system is opaque if the intruder never reveals the secret. In other words, for any secret behavior of the system, there is a non-secret behavior of the system that looks the same to the intruder.

If the secret is modeled as a set of states, the opacity is referred to as state-based. Bryans et al. (2005) introduced state-based opacity for systems modeled by Petri nets, Saboori and Hadjicostis (2007) adapted it to (stochastic) automata, and Bryans et al. (2008) generalized it to transition systems. If the secret is modeled as a set of behaviors, the opacity is referred to as language-based. Language-based opacity was introduced by Badouel et al. (2007) and Dubreil et al. (2008). For more details, we refer the reader to the overview by Jacob et al. (2016).

Several notions of opacity have been introduced in the literature. In this paper, we are interested in the notions of current-state opacity (CSO), initial-state opacity (ISO), initial-and-final-state opacity (IFO), language-based opacity (LBO), K-step opacity (K-SO), and infinite-step opacity (INSO). Current-state opacity is the property that the intruder can never decide whether the system is currently in a secret state. Initial-state opacity is the property that the intruder can never reveal whether the computation started in a secret state. Initial-and-final-state opacity of Wu and Lafortune (2013) is a generalization of both, where the secret is represented as a pair of an initial and a marked state. Consequently, initial-state opacity is a special case of initial-and-final-state opacity where the marked states do not play a role, and current-state opacity is a special case where the initial states do not play a role.

While initial-state opacity prevents the intruder from revealing, at any time during the computation, whether the system started in a secret state, current-state opacity prevents the intruder only from revealing whether the current state of the system is a secret state. However, it may happen that the intruder realizes in the future that the system was in a secret state at some former point of the computation. For instance, if the intruder estimates that the system is in one of two possible states and, in the next step, the system proceeds by an observable event that is possible only from one of the states, then the intruder reveals the state in which the system was one step ago.

This issue has been considered in the literature and led to the notions of K-step opacity (K-SO) and infinite-step opacity (INSO) introduced by Saboori and Hadjicostis (2007, 2012). While K-step opacity requires that the intruder cannot ascertain the secret in the current and K subsequent states, infinite-step opacity requires that the intruder can never ascertain that the system was in a secret state. Notice that 0-step opacity coincides with current-state opacity by definition, and that an n-state automaton is infinite-step opaque if and only if it is $(2^n - 2)$-step opaque (Yin and Lafortune 2017).

Comparing different notions of opacity for automata models, Saboori and Hadjicostis (Saboori and Hadjicostis 2008) provided a language-based definition of initial-state opacity, Cassez et al. (2012) transformed language-based opacity to current-state opacity, and Wu and Lafortune showed that current-state opacity, initial-and-final-state opacity, and language-based opacity can be transformed to each other. They further provided transformations of initial-state opacity to language-based opacity and to initial-and-final-state opacity, and, for prefix-closed languages, a transformation of language-based opacity to initial-state opacity.

In this paper, we extend these results by showing that, for automata models, all the discussed notions of opacity are transformable to each other. As well as the existing transformations, our transformations are computable in polynomial time and preserve the number of observable events and determinism (whenever it is meaningful). In more detail, the transformations of Wu and Lafortune (2013) preserve the determinism of transitions, but result in automata with a set of initial states. This issue can, however, be easily fixed by adding a new initial state, connecting it to the original initial states by new unobservable events, and making the original initial states non-initial. We summarize our results, together with the existing results, in Fig. 1.

There are two immediate applications of the transformations. First, the transformations provide a deeper understanding of the differences among the opacity notions from the structural point of view. For instance, the reader may deduce from the transformations that, for prefix-closed languages, the notions of language-based opacity, initial-state opacity, and current-state opacity coincide, or that to transform current-state opacity to infinite-step opacity means to add only a single state and a few transitions.

Second, the transformations provide a tool to obtain the complexity results for all the discussed opacity notions by studying just one of the notions. For an illustration, consider, for instance, our recent result showing that deciding current-state opacity for systems modeled by DFAs with three events, one of which is unobservable, is PSpace-complete (Balun and Masopust 2020). Since we can transform the problems of deciding current-state opacity and of deciding infinite-step opacity to each other in polynomial time, preserving determinism and the number of observable events, we obtain that deciding infinite-step opacity for systems modeled by DFAs with three events, one of which is unobservable, is PSpace-complete as well. In particular, combining the transformations with known results (Balun and Masopust 2020; Jacob et al. 2016), we obtain a complete complexity picture of the verification of the discussed notions of opacity as summarized in Table 1.

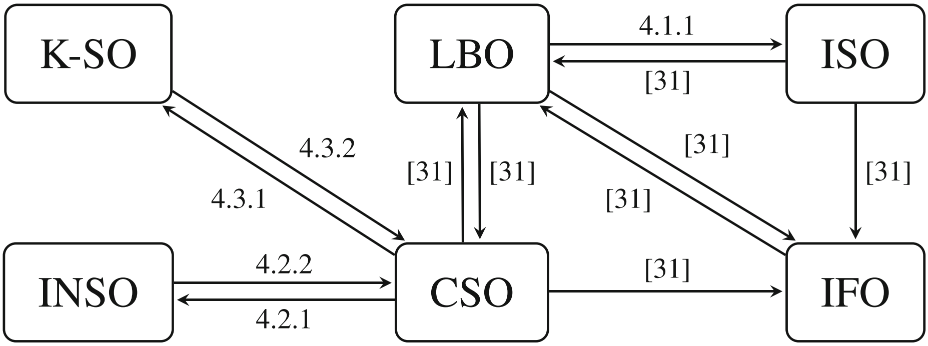


Fig. 1 Overview of the transformations among the notions of opacity for automata models

Table 1 Complexity of verifying the notions of opacity for DESs with Σ_o being the set of observable events following from the transformations and known results; n stands for the number of states of the input automaton, ℓ for the number of observable events of the input automaton, and $m \le \ell n^2$ for the number of transitions in the projected automaton of the input automaton

Opacity notion	$\lvert\Sigma_o\rvert = 1$	$\lvert\Sigma_o\rvert \ge 2$	Order
CSO	coNP-complete (Balun and Masopust 2020)	PSpace-complete (Balun and Masopust 2020)	$O(\ell 2^n)$ (Saboori 2011)
LBO	coNP-complete	PSpace-complete	$O((n + m\ell)2^n)$ (Thm 3)
ISO	NL-complete (Thm 2)	PSpace-complete	$O(\ell 2^n)$ (Wu and Lafortune 2013)
IFO	coNP-complete	PSpace-complete	$O(\ell 2^{2n})$ (Wu and Lafortune 2013)
K-SO	coNP-complete	PSpace-complete	$O((K + 1)2^n(n + m\ell^2))$ (Section 4.3.4)
INSO	coNP-complete	PSpace-complete	$O((n + m\ell)2^n)$ (Section 4.2.4)

The fact that checking opacity for DESs is PSpace-complete was known for some of the considered notions (Jacob et al. 2016). In particular, deciding current-state opacity, initial-state opacity, and language-based opacity were known to be PSpace-complete, deciding K-step opacity was known to be NP-hard, and deciding infinite-step opacity was known to be PSpace-hard.

Complexity theory tells us that any two PSpace-complete problems can be transformed to each other in polynomial time. In other words, it gives the existence of polynomial transformations between the notions of opacity for which the verification is PSpace-complete. However, the theory and the PSpace-hardness proofs presented in the literature do not give a clue how to obtain these transformations. Therefore, from the complexity point of view, our contribution is not the existence of the transformations, but the construction of specific transformations. Since the presented transformations preserve determinism and the number of observable events, they allow us to present stronger results than those known in the literature (Jacob et al. 2016) that we summarize in Table 1.

The transformations further allow us to improve the algorithmic complexity of deciding language-based opacity, infinite-step opacity, and K-step opacity, although we do not use the transformations themselves, but rather the deeper insight into the problems they provide. For language-based opacity, Lin (2011) suggested an algorithm with complexity $O(2^{2n})$, where n is the number of states of the input automaton. In this paper, we improve this complexity to $O((n + m\ell)2^n)$, where $\ell = |\Sigma_o|$ is the number of observable events and $m \le \ell n^2$ is the number of transitions in the projected automaton of the input automaton. For infinite-step opacity and K-step opacity, the latest results are by Yin and Lafortune (2017) who designed an algorithm for checking infinite-step opacity with complexity $O(\ell 2^{2n})$, and an algorithm for checking K-step opacity with complexity $O(\min\{\ell 2^{2n}, \ell^{K+1}2^n\})$. In this paper, we suggest a new algorithm for deciding infinite-step opacity with complexity $O((n + m\ell)2^n)$, and a new algorithm for checking K-step opacity with complexity $O((K + 1)2^n(n + m\ell^2))$. Notice that K is bounded by $2^n - 2$, since an n-state automaton is infinite-step opaque if and only if it is $(2^n - 2)$-step opaque (Yin and Lafortune 2017). Consequently, our algorithm improves the complexity if K is either very large (larger than $2^n - 2$) or polynomial with respect to n; otherwise, the two-way observer technique of Yin and Lafortune (2017) is more efficient, and it is a challenging open problem whether its complexity can be further improved. All our results are summarized in Table 1.

2 Preliminaries

We assume that the reader is familiar with the basic notions of automata theory (Cassandras and Lafortune 2008). For a set S, $|S|$ denotes the cardinality of S, and 2^S the power set of S. Let $\mathbb{N}$ denote the set of all non-negative integers. An alphabet Σ is a finite nonempty set of events. A string over Σ is a sequence of events from Σ. Let Σ^* denote the set of all finite strings over Σ; the empty string is denoted by ε. A language L over Σ is a subset of Σ^*. The set of all prefixes of strings of L is the set $\overline{L} = \{u \mid \text{there is } v \in \Sigma^* \text{ such that } uv \in L\}$. For a string $u \in \Sigma^*$, $|u|$ denotes the length of u, and $\overline{u}$ denotes the set of all prefixes of u.

A *nondeterministic finite automaton* (NFA) over an alphabet Σ is a structure $\mathcal{G} = (Q, \Sigma, \delta, I, F)$, where Q is a finite set of states, $I \subseteq Q$ is a set of initial states, $F \subseteq Q$ is a set of marked states, and $\delta \colon Q \times \Sigma \to 2^Q$ is a transition function that can be extended to the domain $2^Q \times \Sigma^*$ by induction. To simplify our proofs, we use the notation $\delta(Q, S) = \cup_{s\in S} \delta(Q, s)$, where $S \subseteq \Sigma^*$. For a set of states $Q_0 \subseteq Q$, the language marked by $\mathcal{G}$ from the states of Q_0 is the set $L_m(\mathcal{G}, Q_0) = \{w \in \Sigma^* \mid \delta(Q_0, w) \cap F \neq \emptyset\}$, and the language generated by $\mathcal{G}$ from the states of Q_0 is the set $L(\mathcal{G}, Q_0) = \{w \in \Sigma^* \mid \delta(Q_0, w) \neq \emptyset\}$. The language marked by $\mathcal{G}$ is then $L_m(\mathcal{G}) = L_m(\mathcal{G}, I)$, and the language generated by $\mathcal{G}$ is $L(\mathcal{G}) = L(\mathcal{G}, I)$. The NFA $\mathcal{G}$ is *deterministic* (DFA) if $|I| = 1$ and $|\delta(q, a)| \leq 1$ for every $q \in Q$ and $a \in \Sigma$. An automaton $\mathcal{G}$ is *non-blocking* if $\overline{L_m(\mathcal{G})} = L(\mathcal{G})$.

A *discrete-event system* (DES) G over Σ is an NFA together with the partition of the alphabet Σ into two disjoint subsets Σ_o and $\Sigma_{uo} = \Sigma \setminus \Sigma_o$ of *observable* and *unobservable events*, respectively. In the case where all states of the automaton are marked, we simply write $G = (Q, \Sigma, \delta, I)$ without specifying the set of marked states.

When discussing the state estimation properties, the literature often studies deterministic systems with a set of initial states. Such systems are known as deterministic DES and defined as a DFA with several initial states; namely, a *deterministic* DES is an NFA $\mathcal{G} = (Q, \Sigma, \delta, I, F)$, where $|\delta(q, a)| \leq 1$ for every $q \in Q$ and $a \in \Sigma$.

The opacity property is based on partial observations of events described by *projection* $P \colon \Sigma^* \to \Sigma_o^*$. The projection is a morphism defined by $P(a) = \varepsilon$ for $a \in \Sigma_{uo}$, and $P(a) = a$ for $a \in \Sigma_o$. The action of P on a string $\sigma_1\sigma_2 \cdots \sigma_n$, with $\sigma_i \in \Sigma$ for $1 \leq i \leq n$, is to erase all events that do not belong to Σ_o, that is, $P(\sigma_1\sigma_2 \cdots \sigma_n) = P(\sigma_1)P(\sigma_2) \cdots P(\sigma_n)$. The definition can be readily extended to languages.

Let G be a NFA over Σ, and let $P \colon \Sigma^* \to \Sigma_o^*$ be a projection. By the *projected automaton* of G, we mean the automaton $P(G)$ obtained from G by replacing every transition (p, a, q) by the transition $(p, P(a), q)$, and by eliminating the ε-transitions. In particular, if δ is the transition function of G, then the transition function γ of the automaton $P(G)$ is defined as $\gamma(q, a) = \hat{\delta}(q, a)$, where $\hat{\delta} \colon Q \times \Sigma^* \to 2^Q$ is the extension of δ to the domain $Q \times \Sigma^*$, that is, $\hat{\delta}(q, \varepsilon) = \{q\}$ and $\hat{\delta}(q, wa) = \bigcup_{p\in\hat{\delta}(q,w)} \delta(p, a)$. Then $P(G)$ is an NFA over Σ_o, with the same set of states as G, that recognizes the language $P(L_m(G))$ and can be constructed in polynomial time (Hopcroft and Ullman 1979). The DFA constructed from $P(G)$ by the subset construction is called an *observer* (Cassandras and Lafortune 2008). In the worst case, the observer has exponentially many states compared with the automaton G (Jirásková and Masopust 2012; Wong 1998).

A *decision problem* is a yes-no question. A decision problem is *decidable* if there is an algorithm that solves it. Complexity theory classifies decidable problems into classes based on the time or space an algorithm needs to solve the problem. The complexity classes we consider are L, NL, P, NP, and PSPACE denoting the classes of problems solvable by a deterministic logarithmic-space, nondeterministic logarithmic-space, deterministic polynomial-time, nondeterministic polynomial-time, and deterministic polynomial-space algorithm, respectively. The hierarchy of classes is L $\subseteq$ NL $\subseteq$ P $\subseteq$ NP $\subseteq$ PSPACE. Which of the inclusions are strict is an open problem. The widely accepted conjecture is that all are strict. A decision problem is NL-complete (resp. NP-complete, PSPACE-complete) if (i) it belongs to NL (resp. NP, PSPACE) and (ii) every problem from NL (resp. NP, PSPACE) can be reduced to it by a deterministic logarithmic-space (resp. polynomial-time) algorithm. Condition (i) is called *membership* and condition (ii) *hardness*.

3 Notions of opacity

In this section, we recall the definitions of the notions of opacity we discuss. The notion of initial-and-final-state opacity is recalled to make the paper self-contained.

Current-state opacity asks whether the intruder cannot decide, at any instance of time, whether the system is currently in a secret state.

Definition 1 (Current-state opacity (CSO)) Given a DES $G = (Q, \Sigma, \delta, I)$, a projection $P\colon \Sigma^* \to \Sigma_o^*$, a set of secret states $Q_S \subseteq Q$, and a set of non-secret states $Q_{NS} \subseteq Q$. System G is *current-state opaque* if for every string w such that $\delta(I, w) \cap Q_S \neq \emptyset$, there exists a string w' such that $P(w) = P(w')$ and $\delta(I, w') \cap Q_{NS} \neq \emptyset$.

The definition of current-state opacity can be reformulated as a language inclusion as shown in the following lemma. This result is similar to that of Wu and Lafortune (2013) used to transform current-state opacity to language-based opacity. We use this alternative definition to simplify proofs.

Lemma 1 Balun and Masopust (2020) *Let $G = (Q, \Sigma, \delta, I)$ be a DES, $P\colon \Sigma^* \to \Sigma_o^*$ a projection, and $Q_S, Q_{NS} \subseteq Q$ sets of secret and non-secret states, respectively. Let $G_S = (Q, \Sigma, \delta, I, Q_S)$ and $G_{NS} = (Q, \Sigma, \delta, I, Q_{NS})$, then G is current-state opaque if and only if $L_m(P(G_S)) \subseteq L_m(P(G_{NS}))$.*

The second notion of opacity under consideration is language-based opacity. Intuitively, a system is language-based opaque if for any string w in the secret language, there exists a string w' in the non-secret language with the same observation $P(w) = P(w')$. In this case, the intruder cannot conclude whether the secret string w or the non-secret string w' has occurred. We recall the most general definition by Lin (2011).

Definition 2 (Language-based opacity (LBO)) Given a DES $G = (Q, \Sigma, \delta, I)$, a projection $P\colon \Sigma^* \to \Sigma_o^*$, a secret language $L_S \subseteq L(G)$, and a non-secret language $L_{NS} \subseteq L(G)$. System G is *language-based opaque* if $L_S \subseteq P^{-1}P(L_{NS})$.

It is worth mentioning that the secret and non-secret languages are often considered to be regular; and we consider it as well. The reason is that, for non-regular languages, the inclusion problem is undecidable; see Asveld and Nijholt (2000) for more details.

The third notion is the notion of initial-state opacity. Initial-state opacity asks whether the intruder can never reveal whether the computation started in a secret state.

Definition 3 (Initial-state opacity (ISO)) Given a DES $G = (Q, \Sigma, \delta, I)$, a projection $P\colon \Sigma^* \to \Sigma_o^*$, a set of secret initial states $Q_S \subseteq I$, and a set of non-secret initial states $Q_{NS} \subseteq I$. System G is *initial-state opaque* with respect to Q_S, Q_{NS} and P if for every $w \in L(G, Q_S)$, there exists $w' \in L(G, Q_{NS})$ such that $P(w) = P(w')$.

The fourth notion is the notion of initial-and-final-state opacity of Wu and Lafortune (2013). Initial-and-final-state opacity is a generalization of both current-state opacity and initial-state opacity, where the secret is represented as a pair of an initial and a marked state. Consequently, initial-state opacity is a special case of initial-and-final-state opacity where the marked states do not play a role, and current-state opacity is a special case where the initial states do not play a role.

Definition 4 (Initial-and-final-state opacity (IFO)) Given a DES $G = (Q, \Sigma, \delta, I)$, a projection $P\colon \Sigma^* \to \Sigma_o^*$, a set of secret state pairs $Q_S \subseteq I \times Q$, and a set of non-secret state pairs $Q_{NS} \subseteq I \times Q$. System G is *initial-and-final-state opaque* with respect to Q_S, Q_{NS} and P if for every secret pair $(q_0, q_f) \in Q_S$ and every $w \in L(G, q_0)$ such that $q_f \in \delta(q_0, w)$, there exists $(q_0', q_f') \in Q_{NS}$ and $w' \in L(G, q_0')$ such that $q_f' \in \delta(q_0', w')$ and $P(w) = P(w')$.

The fifth notion is the notion of K-step opacity. K-step opacity is a generalization of current-state opacity requiring that the intruder cannot reveal the secret in the current and K subsequent states. By definition, current-state opacity is equivalent to 0-step opacity.

We slightly generalize and reformulate the definition of Saboori and Hadjicostis (2012).

Definition 5 (K-step opacity (K-SO)) Given a system $G = (Q, \Sigma, \delta, I)$, a projection $P\colon \Sigma^* \to \Sigma_o^*$, a set of secret states $Q_S \subseteq Q$, a set of non-secret states $Q_{NS} \subseteq Q$, and a non-negative integer $K \in \mathbb{N}$. System G is *K-step opaque* with respect to Q_S, Q_{NS}, and P if for every string $st \in L(G)$ such that $|P(t)| \le K$ and $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$, there exists a string $s't' \in L(G)$ such that $P(s) = P(s')$, $P(t) = P(t')$, and $\delta(\delta(I, s') \cap Q_{NS}, t') \neq \emptyset$.

Finally, the last notion we consider is the notion of infinite-step opacity. Infinite-step opacity is a further generalization of K-step opacity by setting K being infinity. Actually, Yin and Lafortune (2017) have shown that an n-state automaton is infinite-step opaque if and only if it is $(2^n - 2)$-step opaque. Again, we slightly generalize and reformulate the definition of Saboori and Hadjicostis (Saboori and Hadjicostis 2011).

Definition 6 (Infinite-step opacity (INSO)) Given a system $G = (Q, \Sigma, \delta, I)$, a projection $P\colon \Sigma^* \to \Sigma_o^*$, a set of secret states $Q_S \subseteq Q$, and a set of non-secret states $Q_{NS} \subseteq Q$. System G is *infinite-step opaque* with respect to Q_S, Q_{NS} and P if for every string $st \in L(G)$ such that $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$, there exists a string $s't' \in L(G)$ such that $P(s) = P(s')$, $P(t) = P(t')$, and $\delta(\delta(I, s') \cap Q_{NS}, t') \neq \emptyset$.

4 Transformations

Although some of the transformations were previously known in the literature, Wu and Lafortune (2013) were first who studied the transformations systematically. In particular, they provided polynomial-time transformations among current-state opacity, language-based opacity, initial-state opacity, and initial-and-final-state opacity, see Fig. 1. Inspecting the reductions, it can be seen that after eliminating the unnecessary `Trim` operations, the transformations use only logarithmic space, preserve the number of observable events, and determinism (whenever it is meaningful). As we already pointed out, the transformations of Wu and Lafortune (2013) preserve the determinism of transitions, but they admit a set of initial states. This issue can, however, be easily eliminated by adding a new initial state, connecting it to the original initial states by new unobservable events, and making the original initial states non-initial.

However, their transformation from language-based opacity to initial-state opacity is restricted only to the case where the secret and non-secret languages of the language-based opacity problem are prefix closed. We complete the polynomial-time transformations among all the discussed notions of opacity. In particular, we provide a general transformation from language-based opacity to initial-state opacity in Section 4.1.1, transformations between infinite-step opacity and current-state opacity in Section 4.2, and transformations between K-step opacity and current-state opacity in Section 4.3. All the transformations preserve the number of observable events and determinism. Except for a few exceptions, the transformations need only logarithmic space. Our results are summarized in Fig. 1 with references to the corresponding sections.

The following auxiliary lemma states that we can reduce the number of observable events in DESs with at least three observable events without affecting current-state opacity and initial-state opacity of the DES. We make use of this lemma to preserve the number of observable events in cases where we introduce new observable events in our reductions, namely in Sections 4.1.1, 4.2.2, and 4.3.2.

Lemma 2 *Let $G = (Q, \Sigma, \delta, I, F)$ be an NFA, and let $\Gamma_o \subseteq \Sigma_o$ contain at least three events. Let $G' = (Q', (\Sigma - \Gamma_o) \cup \{0, 1\}, \delta', I, F)$ be an NFA obtained from G as follows. Let $k = \lceil \log_2(|\Gamma_o|) \rceil$, and let $e\colon \Gamma_o \to \{0, 1\}^k$ be a binary encoding of the events of Γ_o. We replace every transition (p, a, q) with $a \in \Gamma_o$ by k transitions*

$$(p, b_1, p_{b_1}), (p_{b_1}, b_2, p_{b_1 b_2}), \dots, (p_{b_1 \cdots b_{k-1}}, b_k, q)$$

where $e(a) = b_1 b_2 \cdots b_k \in \{0, 1\}^k$, and $p_{b_1}, \dots, p_{b_1 \cdots b_{k-1}}$ are states that are added to the state set of G'. Notice that these states are neither secret nor non-secret and that, to preserve determinism, they are newly created when they are needed for the first time, and reused when they are needed later during the replacements, cf. Figure 2 *illustrating a replacement of three observable events $\{a_1, a_2, a_3\}$ with the encoding $e(a_1) = 00$, $e(a_2) = 01$, and $e(a_3) = 10$.*

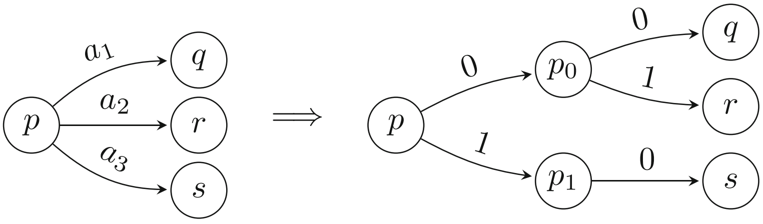


Fig. 2 The replacement of three observable events $\{a_1, a_2, a_3\}$ with the encoding $e(a_1) = 00$, $e(a_2) = 01$, and $e(a_3) = 10$, and new states p_0 and p_1

Then G is current-state (initial-state) opaque with respect to Q_S, Q_{NS}, and $P\colon \Sigma^* \to \Sigma_o^*$ if and only if G' is current-state (initial-state) opaque with respect to Q_S, Q_{NS}, and $P'\colon [(\Sigma - \Gamma_o) \cup \{0,1\}]^* \to [(\Sigma_o - \Gamma_o) \cup \{0,1\}]^*$.

Proof To show that G is current-state opaque if and only if G' is current-state opaque, we define the languages $L_S = L_m(Q, \Sigma, \delta, I, Q_S)$, $L_{NS} = L_m(Q, \Sigma, \delta, I, Q_{NS})$, $L'_S = L_m(Q', (\Sigma - \Gamma_o) \cup \{0,1\}, \delta', I, Q_S)$, and $L'_{NS} = L_m(Q', (\Sigma - \Gamma_o) \cup \{0,1\}, \delta', I, Q_{NS})$. Using Lemma 1, we now need to show that $P(L_S) \subseteq P(L_{NS})$ if and only if $P'(L'_S) \subseteq P'(L'_{NS})$. To this end, we define a morphism $f\colon \Sigma^* \to ((\Sigma - \Gamma_o) \cup \{0,1\})^*$ so that $f(a) = e(a)$ for $a \in \Gamma_o$, and $f(a) = a$ for $a \in \Sigma - \Gamma_o$. By the definition of e and the construction of G', for any string w, we have that $w \in L(G)$ if and only if $f(w) \in L(G')$. In particular, $P(w) \in P(L_S)$ if and only if $P'(f(w)) \in P'(L'_S)$, and $P(w) \in P(L_{NS})$ if and only if $P'(f(w)) \in P'(L'_{NS})$, which completes this part of the proof.

To show that G is initial-state opaque if and only if G' is initial-state opaque, we define the languages $L_S = L(Q, \Sigma, \delta, Q_S)$, $L_{NS} = L(Q, \Sigma, \delta, Q_{NS})$, $L'_S = L(Q', (\Sigma - \Gamma_o) \cup \{0,1\}, \delta', Q_S)$, and $L'_{NS} = L(Q', (\Sigma - \Gamma_o) \cup \{0,1\}, \delta', Q_{NS})$. Since this transforms initial-state opacity to language-based opacity (Wu and Lafortune 2013), it is sufficient to show that $P(L_S) \subseteq P(L_{NS})$ if and only if $P'(L'_S) \subseteq P'(L'_{NS})$. However, this can be shown analogously as above. □

Notice that this binary encoding can be done in polynomial time, and that it preserves determinism.

4.1 Transformations between LBO and ISO

In this section, we discuss the transformations between language-based opacity and initial-state opacity. The transformation from initial-state opacity to language-based opacity has been provided by Wu and Lafortune (2013), as well as the transformation from language-based opacity to initial-state opacity for the case where both the secret and the non-secret language of the language-based opacity problem are prefix closed. We now extend the transformation from language-based opacity to initial-state opacity to the general case.

4.1.1 Transforming LBO to ISO

The language-based opacity problem consists of a DES G_{LBO} over Σ, a projection $P\colon \Sigma^* \to \Sigma_o^*$, a secret language $L_S \subseteq L(G)$, and a non-secret language $L_{NS} \subseteq L(G)$. We transform it to a DES G_{ISO} in such a way that G_{LBO} is language-based opaque if and only if G_{ISO} is initial-state opaque.

Assume that the languages L_S and L_{NS} are represented by the non-blocking automata $A_S = (Q_S, \Sigma_S, \delta_S, I_S, F_S)$ and $A_{NS} = (Q_{NS}, \Sigma_{NS}, \delta_{NS}, I_{NS}, F_{NS})$, respectively. Without loss of generality, we may assume that their sets of states are disjoint, that is, $Q_S \cap Q_{NS} = \emptyset$.

Our transformation proceeds in two steps:

1. We construct a DES G_{ISO} with one additional observable event @.
2. We use Lemma 2 to reduce the number of observable events of G_{ISO} by one.

Since the second step follows from Lemma 2, we only describe the first step, that is, the construction of G_{ISO} over $\Sigma \cup \{@\}$, and the specification of the sets of secret states Q'_S and non-secret states Q'_{NS}.

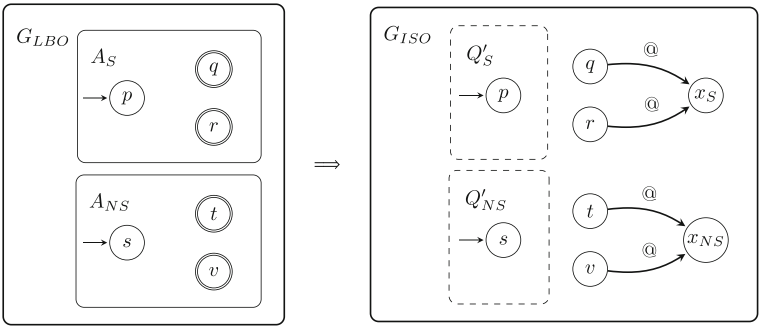


Fig. 3 Transforming LBO to ISO

From the automata A_S and A_{NS}, we construct the automata $G_S = (Q_S \cup \{x_S\}, \Sigma_S, \delta_S, I_S, Q_S \cup \{x_S\})$ and $G_{NS} = (Q_{NS} \cup \{x_{NS}\}, \Sigma_{NS}, \delta_{NS}, I_{NS}, Q_{NS} \cup \{x_{NS}\})$ by adding two new states x_S and x_{NS}, and the following transitions, see Fig. 3 for an illustration of the construction:

- for every state $q \in F_S$, we add a new transition $(q, @, x_S)$ to δ_S;
- for every state $q \in F_{NS}$, we add a new transition $(q, @, x_{NS})$ to δ_{NS}.

Let $Q'_S = I_S$ denote the set of secret initial states of G_{ISO}, and let $Q'_{NS} = I_{NS}$ denote the set of non-secret initial states of G_{ISO}. We extend projection P to $P' \colon (\Sigma \cup \{@\})^* \to (\Sigma_o \cup \{@\})^*$. Finally, let G_{ISO} denote the automata G_S and G_{NS} considered as a single NFA. Before we show that G_{LBO} is language-based opaque if and only if G_{ISO} is initial-state opaque, notice that the transformation can be done in polynomial time and that it preserves determinism.

Theorem 1 *The DES G_{LBO} is language-based opaque with respect to L_S, L_{NS}, and P if and only if the DES G_{ISO} is initial-state opaque with respect to Q'_S, Q'_{NS}, and P'.*

Proof We need to show that $P(L_S) \subseteq P(L_{NS})$ if and only if $P'(L(G_S)) \subseteq P'(L(G_{NS}))$.

However, by construction, $L(G_S) = \overline{L_S} \cup L_S@$ and $L(G_{NS}) = \overline{L_{NS}} \cup L_{NS}@$, and hence $P(L_S) \subseteq P(L_{NS})$ if and only if $P'(L(G_S)) \subseteq P'(L(G_{NS}))$, which is if and only if G_{ISO} is initial-state opaque. □

We now provide an illustrative example.

Example 1 Let G_1 over $\Sigma = \{a, b, c\}$ depicted in Fig. 4 (left) be the instance of the LBO problem with the secret language $L_S = abb^*$ and the non-secret language $L_{NS} = acb^*$. Our transformation of LBO to ISO then results in the DES G'_1 depicted in Fig. 4 (right) with a new observable event @, a single secret initial state 1, and a single non-secret initial state 4. We distinguish two cases depending on whether event c is observable or not.

In the first case, we assume that event c is unobservable. In this case, G_1 is language-based opaque, because $P(L_S) \subseteq P(L_{NS})$, and the reader can see that $P(L(G'_1, 1)) = \overline{abb^*@} \subseteq \overline{ab^*@} = P(L(G'_1, 4))$. Therefore, G'_1 is initial-state opaque.

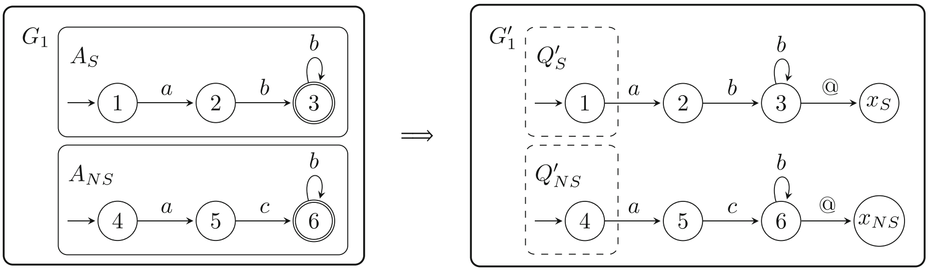


Fig. 4 An example of the transformation of the LBO problem (left) to the ISO problem (right)

In the second case, we assume that event c is observable. In this case, G_1 is not language-based opaque, because $ab \in P(L_S)$ whereas $ab \notin P(L_{NS})$, and we can see that $ab \in L(G'_1, 1)$ and $ab \notin L(G'_1, 4)$. Therefore, G'_1 is not initial-state opaque.

4.1.2 The case of a single observable event

The second step of our construction, Lemma 2, requires that G_{ISO} has at least three observable events or, equivalently, that G_{LBO} has at least two observable events. Consequently, our transformation does not preserve the number of observable events if G_{LBO} has a single observable event. In fact, we show that there does not exist such a transformation unless P = NP, which is a longstanding open problem of computer science. Deciding language-based opacity for systems with a single observable event is coNP-complete (Holzer and Kutrib 2011; Stockmeyer and Meyer 1973). We show that deciding initial-state opacity for systems with a single observable event is NL-complete, and hence efficiently solvable on a parallel computer (Arora and Barak 2009). In particular, the problem can be solved in polynomial time.

Theorem 2 *Deciding initial-state opacity for DESs with a single observable event is* NL-*complete.*

Proof Deciding initial-state opacity is equivalent to checking the inclusion of two prefix-closed languages. Namely, a DES G with $\Sigma_o = \{a\}$ is initial-state opaque with respect to secret states Q_S and non-secret states Q_{NS} if and only if $K_S \subseteq K_{NS}$ for $K_S = P(L(G, Q_S))$ and $K_{NS} = P(L(G, Q_{NS}))$. Since the languages K_S and K_{NS} are prefix-closed, they are either finite, consisting of at most $|Q|$ strings, or equal to $\{a\}^*$.

To show that the problem belongs to NL, we show how to verify $K_S \not\subseteq K_{NS}$ in nondeterministic logarithmic space. Then, since NL is closed under complement (Immerman 1988; Szelepcsėnyi 1988), $K_S \subseteq K_{NS}$ belongs to NL. Thus, to check that $K_S \not\subseteq K_{NS}$ in nondeterministic logarithmic space, we guess $k \in \{0, \dots, |Q|\}$ in binary, store it in logarithmic space, and verify that $a^k \in K_S$ and $a^k \notin K_{NS}$. To verify $a^k \in K_S$, we guess a path in G step by step, storing only the current state, and counting the number of steps by decreasing k by one in each step; logarithmic space is sufficient for this. Since $a^k \notin K_{NS}$ belongs to the complement of NL, which coincides with NL, we can check $a^k \notin K_{NS}$ in nondeterministic logarithmic space as well.

To show that deciding initial-state opacity for DESs with a single observable event is NL-hard, we reduce the DAG reachability problem (Jones 1975): given a DAG $G = (V, E)$ and nodes $s, t \in V$, the problem asks whether t is reachable from s.

From G, we construct a DES $\mathcal{A} = (V \cup \{i\}, \{a\}, \delta, \{s, i\})$, where i is a new initial state and a is an observable event, as follows. With each node of G, we associate a state in $\mathcal{A}$. Whenever there is an edge from j to k in G, we add an a-transition from j to k to $\mathcal{A}$. We add a self-loop labeled by a to state t and to state i. The set of secret initial states is $Q_S = \{i\}$ and the set of non-secret initial states $Q_{NS} = \{s\}$. Then, $\mathcal{A}$ is initial-state opaque if and only if there is a path from s to t in G. Indeed, $L(\mathcal{A}, i) = \{a\}^*$ is included in $L(\mathcal{A}, s)$ if and only if $L(\mathcal{A}, s) = \{a\}^*$, which is if and only if t is reachable from s. □

4.1.3 Algorithmic complexity of deciding LBO

The algorithmic complexity of deciding whether a given DES is language-based opaque with respect to given secret and non-secret languages has been investigated in the literature. Lin (2011) suggested an algorithm with the complexity $O(2^{2n})$, where n is the order of the state spaces of the automata representing the secret and non-secret languages. The same complexity has been achieved by Wu and Lafortune (2013) using the transformation to current-state opacity. We improve this complexity.

Theorem 3 *The time complexity of deciding whether a DES G is language-based opaque with respect to a projection P, a secret language $L_S \subseteq L(G)$, and a non-secret language $L_{NS} \subseteq L(G)$ is $O(m\ell 2^{n_2} + n_1 2^{n_2})$, where n_1 is the number of states of the automaton recognizing L_S, n_2 is the number of states recognizing L_{NS}, $m \leq \ell n_1^2$ is the number of transitions of an NFA recognizing $P(L_S)$, and ℓ is the number of observable events.*

Proof Let G_S and G_{NS} be automata recognizing L_S and L_{NS} with n_1 and n_2 states, respectively. Then $P(L_S) \subseteq P(L_{NS})$ if and only if $P(L_S) \cap \text{co-}P(L_{NS}) = \emptyset$, where co-$P(L_{NS})$ stands for $\Sigma^* - P(L_{NS})$. We represent $P(L_S)$ by the projected automaton $P(G_S)$ with m transitions and at most n_1 states, and co-$P(L_{NS})$ by the complement of the observer of G_{NS}, denoted by co-G_{NS}^{obs}, which has at most 2^{n_2} states and $\ell 2^{n_2}$ transitions. The problem is now equivalent to checking whether the language of $P(G_S) \cap \text{co-}G_{NS}^{obs}$ is empty, which means to search the structure for a reachable marked state. Since $P(G_S)$ has at most n_1 states and $m \leq \ell n_1^2$ transitions, the structure has $O(m\ell 2^{n_2} + n_1 2^{n_2})$ transitions and states, which completes the proof. □

4.2 Transformations between CSO and INSO

In this section, we provide the transformations between current-state opacity and infinite-step opacity. To the best of our knowledge, no transformations between current-state opacity and infinite-step opacity have been discussed in the literature so far.

4.2.1 Transforming CSO to INSO

We first focus on the transformation from current-state opacity to infinite-step opacity. The problem of deciding current-state opacity consists of a DES $G_{CSO} = (Q, \Sigma, \delta, I)$, a projection $P \colon \Sigma^* \to \Sigma_o^*$, a set of secret states $Q_S \subseteq Q$, and a set of non-secret states $Q_{NS} \subseteq Q$. From G_{CSO}, we construct a DES G_{INSO} over the alphabet $\Sigma \cup \{u\}$, where u is a new unobservable event. Specifically, we construct $G_{INSO} = (Q \cup \{q^\star\}, \Sigma \cup \{u\}, \delta', I)$ from G_{CSO}

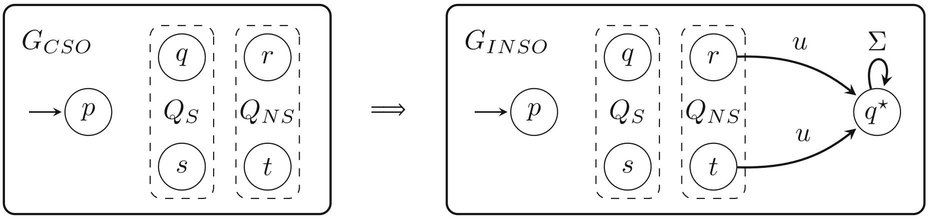


Fig. 5 Transforming CSO to INSO

by adding a new state $q^\star$ that is neither secret nor non-secret, and by defining δ' as follows, see Fig. 5 for an illustration:

1. $\delta' = \delta$, that is, δ' is initialized as δ and further extended as follows;
2. for each state $q \in Q_{NS}$, we add a transition $(q, u, q^\star)$ to δ';
3. for each $a \in \Sigma$, we add a self-loop $(q^\star, a, q^\star)$ to δ'.

We extend the projection P to the projection $P' \colon (\Sigma \cup \{u\})^* \to \Sigma_o^*$. The sets Q_S and Q_{NS} remain unchanged.

Notice that the transformation preserves the number of observable events and determinism, and that it requires only logarithmic space. It remains to show that G_{CSO} is current-state opaque if and only if G_{INSO} is infinite-step opaque.

Theorem 4 *The DES G_{CSO} is current-state opaque with respect to Q_S, Q_{NS}, and P if and only if the DES G_{INSO} is infinite-step opaque with respect to Q_S, Q_{NS}, and P'.*

Proof Assume first that G_{CSO} is not current-state opaque. Since the new state $q^\star$ is neither secret nor non-secret, we have that G_{INSO} is not current-state opaque either. Consequently, G_{INSO} is not infinite-step opaque.

On the other hand, assume that G_{CSO} is current-state opaque. Since the new state $q^\star$ is neither secret nor non-secret, we have that G_{INSO} is current-state opaque as well. Let $st \in L(G_{INSO})$ be such that $\delta'(\delta'(I, s) \cap Q_S, t) \neq \emptyset$; in particular, $\delta'(I, s) \cap Q_S \neq \emptyset$. Then, since G_{INSO} is current-state opaque, there exists $s' \in L(G_{INSO})$ such that $P'(s') = P'(s)$ and $\delta'(I, s') \cap Q_{NS} \neq \emptyset$. By construction, s' can be extended by the string ut using the transitions to state $q^\star$ followed by self-loops in state $q^\star$. Therefore, $\delta'(\delta'(I, s') \cap Q_{NS}, ut) \neq \emptyset$ and $P'(st) = P'(sut)$, which shows that G_{INSO} is infinite-step opaque. □

We now illustrate the construction in the following example.

Example 2 Let G_2 over $\Sigma = \{a, b, c\}$ depicted in Fig. 6 (left) be the instance of the CSO problem with the set of secret states $Q_S = \{2\}$ and the set of non-secret states $Q_{NS} = \{5\}$. Our transformation of CSO to INSO then results in the DES G_2' depicted in Fig. 6 (right) with a new state $q^\star$ and a new unobservable event u. We distinguish two cases depending on whether event c is observable or not.

If event c is unobservable, then G_2 is current-state opaque, because the only string leading to a secret state, state 2, is the string a, for which the string ac leading to the non-secret state, state 5, satisfies that $P(a) = P(ac)$. Then, the reader can see that G_2' is infinite-step opaque, because the only possible extensions of the string a from the secret state 2 are of the form b^k, for $k \in \mathbb{N}$, and for every such extension there is an extension ub^k of the string ac from the non-secret state 5 such that $P(ab^k) = P(acub^k)$.

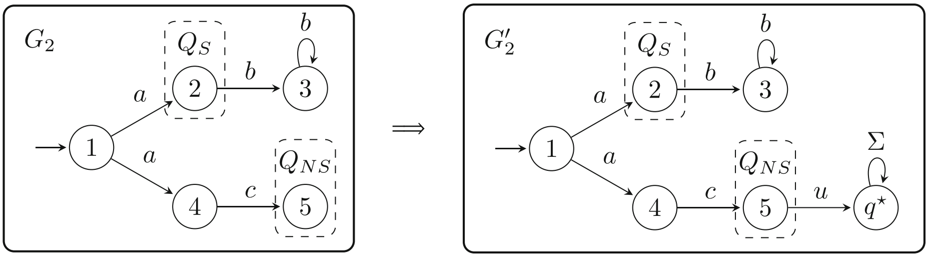


Fig. 6 An example of the transformation of the CSO problem (left) to the INSO problem (right)

If event c is observable, then G_2 is not current-state opaque, because the only string leading to a non-secret state, string ac, has a different observation then the string a leading to the secret state, that is, $P(ac) \neq P(a)$. Consequently, the reader can verify that G_2' is not current-state opaque, and hence neither infinite-step opaque.

4.2.2 Transforming INSO to CSO

Transforming infinite-step opacity to current-state opacity is technically more involved. The problem of deciding infinite-step opacity consists of a DES $G_{INSO} = (Q, \Sigma, \delta, I)$, a projection $P \colon \Sigma^* \to \Sigma_o^*$, a set of secret states $Q_S \subseteq Q$, and a set of non-secret states $Q_{NS} \subseteq Q$. From G_{INSO}, we construct a DES G_{CSO} in the following two steps:

1. We construct a DES G_{CSO} such that G_{CSO} is current-state opaque if and only if G_{INSO} is infinite-step opaque. In this step of the construction, G_{CSO} has one observable event more than G_{INSO}.
2. To reduce the number of observable events by one, we apply Lemma 2. Consequently, the resulting DES has the same number of observable events as G_{INSO}, if G_{INSO} has at least two observable events, is deterministic if and only if G_{CSO} is, and is current-state opaque if and only if G_{CSO} is.

We now describe the construction of $G_{CSO} = (Q \cup Q^+ \cup Q^-, \Sigma \cup \{@\}, \delta', I)$, where $Q^+ = \{q^+ \mid q \in Q\}$, $Q^- = \{q^- \mid q \in Q\}$, and @ is a new observable event. To this end, we first make two disjoint copies of G_{INSO}, denoted by G_S and G_{NS}, where the set of states of G_S is denoted by $Q_S' = Q^+$ and the set of states of G_{NS} is denoted by $Q_{NS}' = Q^-$. The DES G_{CSO} is taken as the disjoint union of the automata G_{INSO}, G_S, and G_{NS}, see Fig. 7 for an illustration. Furthermore, for every state $q \in Q_S$, we add the transition $(q, @, q^+)$ and, for every state $q \in Q_{NS}$, we add the transition $(q, @, q^-)$. The set of secret states of G_{CSO} is Q_S' and the set of non-secret states of G_{CSO} is Q_{NS}'. We extend projection P to $P' \colon (\Sigma \cup \{@\})^* \to (\Sigma_o \cup \{@\})^*$.

Notice that G_{CSO} is deterministic if and only if G_{INSO} is, and that logarithmic space is sufficient for the construction of G_{CSO}. As already pointed out, however, the construction does not preserve the number of observable events, which requires the second step of the construction using Lemma 2 as described above.

We now show that G_{INSO} is infinite-step opaque if and only if G_{CSO} is current-state opaque.

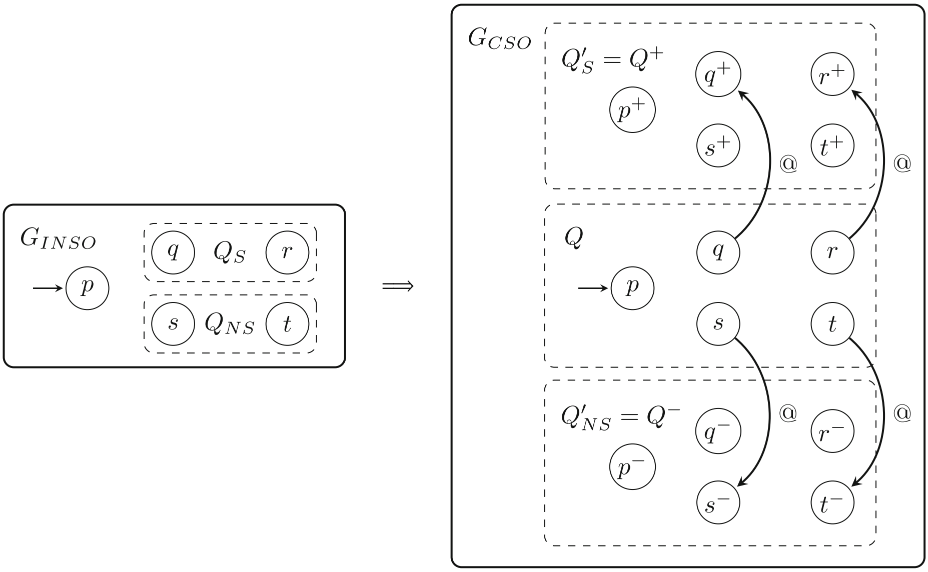


Fig. 7 Transforming INSO to CSO

Theorem 5 *The DES G_{INSO} is infinite-step opaque with respect to Q_S, Q_{NS}, and P if and only if the DES G_{CSO} is current-state opaque with respect to Q'_S, Q'_{NS}, and $P' : (\Sigma \cup \{@\})^* \to (\Sigma_o \cup \{@\})^*$.*

Proof Assume that G_{INSO} is infinite-step opaque. We show that G_{CSO} is current-state opaque. To this end, consider a string w such that $\delta'(I, w) \cap Q'_S \neq \emptyset$. We want to show that there exists w' such that $P'(w) = P'(w')$ and $\delta'(I, w') \cap Q'_{NS} \neq \emptyset$. However, since $Q'_S = Q^+$, w is of the form $w_1 @ w_2$. Then, by the construction, $\delta(I, w_1)$ contains a secret state of G_{INSO}, say $q \in \delta(I, w_1) \cap Q_S$, such that state q^+ is a copy of state q reached under @ from state q in G_{CSO}, and w_2 is read from state q^+ in the copy G_S of G_{INSO}. That is, w_2 can be read from state q in G_{INSO}, and hence $\delta(I, w_1 w_2) \neq \emptyset$. Altogether, $\delta(\delta(I, w_1) \cap Q_S, w_2) \neq \emptyset$ and the fact that G_{INSO} is infinite-step opaque imply that there exists a string $w'_1 w'_2 \in L(G_{INSO})$ such that $P(w_1) = P(w'_1)$, $P(w_2) = P(w'_2)$, and $\delta(\delta(I, w'_1) \cap Q_{NS}, w'_2) \neq \emptyset$. Let $w' = w'_1 @ w'_2$. Then $P'(w) = P'(w')$ and, by the construction, $\emptyset \neq \delta'(\delta'(I, w'_1 @) \cap Q'_{NS}, w'_2) \subseteq Q'_{NS}$, which completes the proof.

On the other hand, assume that G_{INSO} is not infinite-step opaque, that is, there exists a string $st \in L(G_{INSO})$ such that $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$ and for every $s't' \in L(G_{INSO})$ with $P(s) = P(s')$ and $P(t) = P(t')$, $\delta(\delta(I, s') \cap Q_{NS}, t') = \emptyset$. But then for $s@t \in L(G_{CSO})$, we have that $\emptyset \neq \delta'(\delta'(I, s@) \cap Q'_S, t) = \delta'(I, s@t) \subseteq Q'_S$ and, for every $s'@t' \in L(G_{CSO})$ such that $P'(s@t) = P'(s'@t')$, we have that $\delta'(I, s'@t') \cap Q'_{NS} = \delta'(\delta'(I, s'@) \cap Q'_{NS}, t') = \emptyset$, which shows that G_{CSO} is not current-state opaque. □

We now illustrate the construction.

Example 3 Let G_3 over $\Sigma = \{a, b, c\}$ depicted in Fig. 8 (left) be the instance of the INSO problem with the set of secret states $Q_S = \{2\}$ and the set of non-secret states $Q_{NS} = \{4\}$.

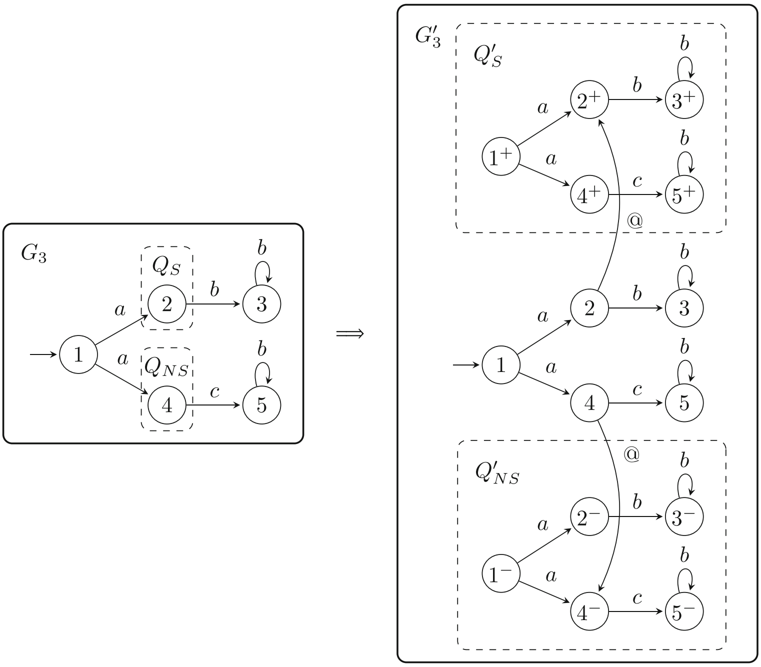


Fig. 8 An example of the transformation of the INSO problem (left) to the CSO problem (right)

Our transformation of INSO to CSO then results in the DES G'_3 depicted in Fig. 8 (right) with a new observable event @, the set of secret states Q'_S, and the set of non-secret states Q'_{NS}. We again consider two cases based on the observability status of event c.

If event c is unobservable, then G_3 is infinite-step opaque. Indeed, the only string leading to the single secret state, state 2, is the string a. The same string leads to the single non-secret state, state 4. Then, any possible extension of the string a from the unique secret state 2 is the string b^k, for $k \in \mathbb{N}$, which reaches state 3. However, for any such extension, there is the extension cb^k from the non-secret state 4 with $P(ab^k) = P(acb^k)$. The reader can further see that G'_3 is current-state opaque, because it can enter a secret state only after generating a string of the form $a@b^k$, $k \in \mathbb{N}$, in which case $\delta'(1, P^{-1}(a@)) = \{2^+, 4^-, 5^-\}$ and $\delta'(1, P^{-1}(a@b^k)) = \{3^+, 5^-\}$ for $k \geq 1$.

If event c is observable, then G_3 is not infinite-step opaque, because after generating string ab, the intruder can deduce that the system was in the secret state 2. Similarly, after generating string $a@b$, system G'_3 ends up in the only state 3^+, which is a secret state, and hence G'_3 is not current-state opaque.

4.2.3 The case of a single observable event

To preserve the number of observable events, our transformation of infinite-step opacity to current state opacity relies on Lemma 2. This lemma requires at least two observable events in G_{INSO}, and hence it is not applicable to systems with a single observable event. For these systems, we provide a different transformation that requires to add at most a quadratic number of new states.

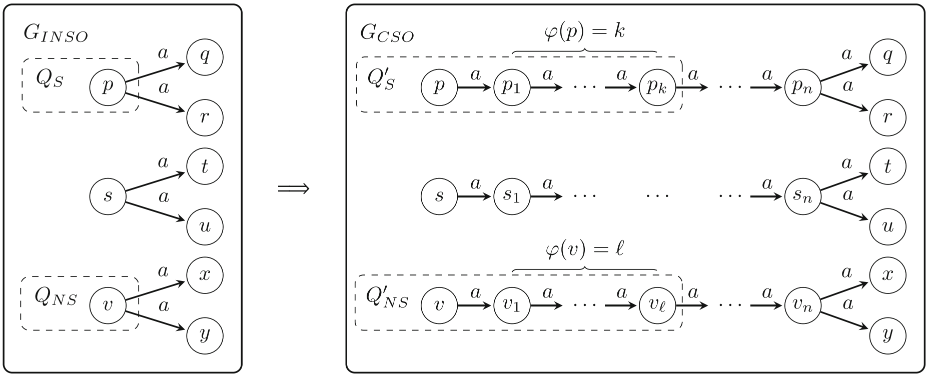


Fig. 9 Transforming INSO to CSO for systems with a single observable event

The problem of deciding infinite-step opacity for systems with a single observable event consists of a DES $G_{INSO} = (Q, \Sigma, \delta, I)$ with $\Sigma_o = \{a\}$, a set of secret states $Q_S \subseteq Q$, a set of non-secret states $Q_{NS} \subseteq Q$, and a projection $P\colon \Sigma^* \to \{a\}^*$.

We denote the number of states of G_{INSO} by n, and define a function $\varphi\colon Q \to \{0, \dots, n\}$ that assigns, to every state q, the maximal number $k \in \{0, \dots, n\}$ of observable steps that are possible from state q; formally, $\varphi(q) = \max \left\{k \in \{0, \dots, n\} \mid \delta(q, P^{-1}(a^k)) \neq \emptyset\right\}$.

From G_{INSO}, we construct a DES $G_{CSO} = (Q', \Sigma, \delta', I)$ as illustrated in Fig. 9, where δ' is initialized as δ and modified as follows. For every state $p \in Q$ with $\varphi(p) > 0$, we add n new states $p_1, \dots, p_n$ to Q' and n new transitions (p, a, p_1) and (p_i, a, p_{i+1}), for $i = 1, \dots, n-1$, to δ'. Finally, we replace every transition (p, a, r) in δ' by the transition (p_n, a, r). Notice that the transformation requires to add at most n^2 states, and hence it can be done in polynomial time. Let $Q'_S = Q_S$ and $Q'_{NS} = Q_{NS}$. For every state $p \in Q_S$ with $\varphi(p) = k > 0$, we add the corresponding states $p_1, \dots, p_k$ to Q'_S. Analogously, for $p \in Q_{NS}$ with $\varphi(p) = k > 0$, we add $p_1, \dots, p_k$ to Q'_{NS}.

Notice that the transformation can be done in polynomial time, preserves the number of observable events, and determinism. However, whether the transformation can be done in logarithmic space is open. Even if the DES had no unobservable event, to determine whether $\varphi(\cdot) = n$ is equivalent to the detection of a cycle. The detection of a cycle is NL-hard: We can reduce the DAG reachability problem as follows. Given a DAG G and two nodes s and t, we construct a DES $\mathcal{G}$ by associating a state with every node of G and an a-transition with every edge of G. Finally, we add an a-transition from t to s. Then t is reachable from s in G if and only if $\mathcal{G}$ contains a cycle. Since it is an open problem whether L = NL, it is an open problem whether φ can be computed in deterministic logarithmic space.

We show that G_{INSO} is infinite-step opaque if and only if G_{CSO} is current-state opaque.

Theorem 6 *The DES G_{INSO} with a single observable event is infinite-step opaque with respect to Q_S, Q_{NS}, and P if and only if the DES G_{CSO} is current-state opaque with respect to Q'_S, Q'_{NS}, and P.*

Proof Assume that G_{INSO} is not infinite-step opaque. Then, there exists $st \in L(G_{INSO})$ with $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$ such that $\delta(\delta(I, P^{-1}P(s)) \cap Q_{NS}, P^{-1}P(t)) = \emptyset$. Let $f\colon \Sigma^* \to \Sigma^*$ be a morphism such that $f(a) = a^{n+1}$ and $f(b) = b$, for $a \neq b \in$

Σ. Then, by construction, $\delta(I, s) = \delta'(I, f(s))$, and hence $\delta'(I, f(s)) \cap Q'_S \neq \emptyset$. If $\delta(I, P^{-1}P(s)) \cap Q_{NS} = \emptyset$, then $\delta(I, f(P^{-1}P(s))) \cap Q'_{NS} = \emptyset$ because $\delta(I, s') = \delta'(I, f(s'))$ for any $s' \in P^{-1}P(s)$, and G_{CSO} is not current-state opaque. Otherwise, we denote by $q_s \in \delta(I, s) \cap Q_S$ and $q_{ns} \in \delta(I, P^{-1}P(s)) \cap Q_{NS}$ the states with maximal $\varphi(q_s)$ and $\varphi(q_{ns})$. Since G_{INSO} is not infinite-step opaque, $\varphi(q_s) > \varphi(q_{ns})$. Then, in G_{CSO}, q_s has exactly one outgoing observable transition and is followed by $\varphi(q_s) = k$ secret states, while q_{ns} is followed by $\varphi(q_{ns}) < k$ non-secret states. Therefore, $\delta'(I, f(s)a^k) \cap Q'_S \neq \emptyset$ and $\delta'(I, f(s')a^k) \cap Q'_{NS} = \emptyset$ for any $s' \in P^{-1}P(s)$, and hence G_{CSO} is not current-state opaque.

On the other hand, assume that G_{INSO} is infinite-step opaque, and that $\delta'(I, w) \cap Q'_S \neq \emptyset$. We show that $\delta'(I, P^{-1}P(w)) \cap Q_{NS} \neq \emptyset$. Consider a state $q_s \in \delta'(I, w) \cap Q'_S$ and a path π in G_{CSO} leading to q_s under w. Denote by p the last state of π that corresponds to a state of G_{INSO}; that is, p is not a new state added by the construction of G_{CSO}. Since $q_s \in Q'_S$, we have, by construction, that $p \in Q_S$. Then the choice of p partitions $w = uv$, where u, read along the path π, leads to state p, and $v = a^\ell$ is a suffix of length $\ell \leq n$. Let u' be a string such that $f(u') = u$. Then $p \in \delta(I, u') \cap Q_S$. Since $\varphi(p) \geq \ell$, there exists t such that $P(t) = a^\ell$ and $\delta(\delta(I, u') \cap Q_S, t) \neq \emptyset$ in G_{INSO}. Then infinite-step opacity of G_{INSO} implies that there exists u'' and t' such that $P(u') = P(u'')$, $P(t) = P(t')$, and $\delta(\delta(I, u'') \cap Q_{NS}, t') \neq \emptyset$. In particular, there is a state $q_{ns} \in \delta(I, u'') \cap Q_{NS}$ with $\varphi(q_{ns}) \geq \ell$, and $\delta'(I, f(u'')) \cap Q'_{NS} \neq \emptyset$. Therefore, $\delta'(I, f(u'')a^\ell) \cap Q'_{NS} \neq \emptyset$ and $P(f(u'')a^\ell) = P(uv) = P(w)$, which completes the proof. □

We now illustrate the construction.

Example 4 Let G_4 over $\Sigma = \{a, u\}$ depicted in Fig. 10 (left) be the instance of the INSO problem with a single observable event $\Sigma_o = \{a\}$, the set of secret states $Q_S = \{1\}$, and the set of non-secret states $Q_{NS} = \{3\}$. Then, $\varphi(1) = \varphi(3) = 3$, and our transformation of INSO to CSO results in the DES G'_4 depicted in Fig. 10 (right) with the set of secret states Q'_S and the set of non-secret states Q'_{NS}. We consider two cases based on the presence of the unobservable transition $(1, u, 3)$ in G_4.

We first assume that the transition $(1, u, 3)$ exists in G_4. Then, G_4 is infinite-step opaque, because any string a^k leading from the secret state 1 is indistinguishable from the string ua^k that leads the system to the non-secret state 3. The reader can see that G'_4 is current-state opaque, because a secret state is reachable only under a string of the form a^k, for $k \in \{0, 1, 2, 3\}$, and for any such string there is an indistinguishable string ua^k reaching a non-secret state.

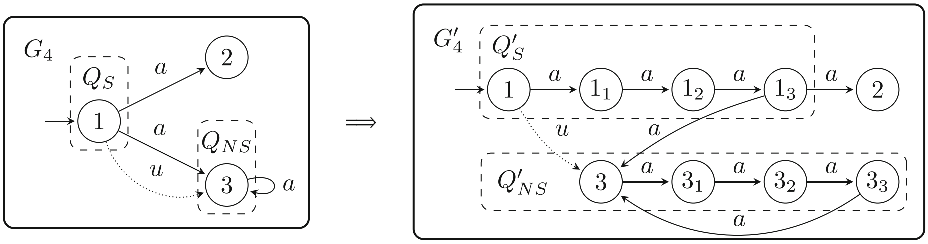


Fig. 10 An example of the transformation of the INSO problem with a single observable event (left) to the CSO problem (right)

If the transition $(1, u, 3)$ does not exist in G_4, then G_4 is not infinite-step opaque, because it is neither current-state opaque and, obviously, neither G_4' is current-state opaque.

4.2.4 Improving the algorithmic complexity of deciding infinite-step opacity

Let $G = (Q, \Sigma, \delta, I, F)$ be a DES. We design an algorithm deciding infinite-step opacity in time $O((n+m\ell)2^n)$, where $\ell = |\Sigma_o|$ is the number of observable events, n is the number of states of G, and m is the number of transitions of $P(G)$, $m \le \ell n^2$.

To decide whether G is infinite-step opaque with respect to $Q_S, Q_{NS} \subseteq Q$, and $P\colon \Sigma^* \to \Sigma_o^*$, we proceed as follows:

1. We compute the observer $\mathcal{G}^{obs}$ of G in time $O(\ell 2^n)$ (Cassandras and Lafortune 2008);
2. We compute the projected automaton $P(G)$ of G in time $O(m + n)$ (Hopcroft and Ullman 1979);
3. We compute the product automaton $\mathcal{C} = P(G) \times \mathcal{G}^{obs}$ in time $O((m + n) \cdot \ell 2^n)$ (Domaratzki and Salomaa 2007); – states of $\mathcal{C}$ are of the form $Q \times 2^Q$;
4. For every reachable state X of $\mathcal{G}^{obs}$, we compute $X_S = X \cap Q_S$ and $X_{NS} = X \cap Q_{NS}$;
   (a) If $X_S \neq \emptyset$ and $X_{NS} = \emptyset$, then G is not infinite-step opaque; this is, actually, the standard check whether G is current-state opaque;
   (b) Otherwise, for every state $x \in X_S$, we add a transition from X under @ to state (x, X_{NS}) of $\mathcal{C}$, and we add the state (x, X_{NS}) to set Y;
5. If $\mathcal{C}$ contains a state of the form $(q, \emptyset)$ reachable from Y, then G is not infinite-step opaque; otherwise, G is infinite-step opaque.

Informally, we first make use of the standard check in the observer of G whether G is current-state opaque. If it is not, then it is neither infinite-step opaque. Otherwise, for every state X of the observer of G that contains both secret and non-secret states, we add a transition under the new event @ to a pair of a secret state $x \in X$ and the set of all non-secret states X_{NS} of X. If a state of the form $(q, \emptyset)$ is reachable from (x, X_{NS}), then G is not infinite-step opaque. Otherwise, G is infinite-step opaque. We now formally prove correctness.

Lemma 3 *The DES G is infinite-step opaque if and only if G is current-state opaque and no state of the form $(q, \emptyset)$ is reachable in $\mathcal{C}$ from the set Y.*

Proof Assume that G is not infinite-step opaque. Then, there exists $st \in L(G)$ such that $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$ and $\delta(\delta(I, P^{-1}P(s)) \cap Q_{NS}, P^{-1}P(t)) = \emptyset$. There are two cases:

(i) either $\delta(I, P^{-1}P(s)) \cap Q_{NS} = \emptyset$, in which case G is not current-state opaque, neither infinite-step opaque, and the algorithm detects this situation in the observer of G on line 4(a),

(ii) or $\delta(I, P^{-1}P(s)) \cap Q_{NS} = Z \neq \emptyset$. In this case, $P(s)$@ leads from the observer of G to the pairs $(\delta(I, P^{-1}P(s)) \cap Q_S) \times \{Z\}$ of the NFA $\mathcal{C}$. Since $\delta(I, st) \neq \emptyset$, there exists $(z, Z) \in (\delta(I, P^{-1}P(s)) \cap Q_S) \times \{Z\}$ such that $P(t)$ leads the projected automaton $P(G)$ from state z to a state q. However, $\delta(Z, P^{-1}P(t)) = \emptyset$ implies that $P(t)$ leads the observer of G from state Z to state $\emptyset$, and hence the pair $(q, \emptyset)$ is reachable in $\mathcal{C}$ from a state of Y.

On the other hand, if G is infinite-step opaque, then it is current-state opaque, and we show that no state of the form $(q, \emptyset)$ is reachable in $\mathcal{C}$ from a state of Y. For the sake of

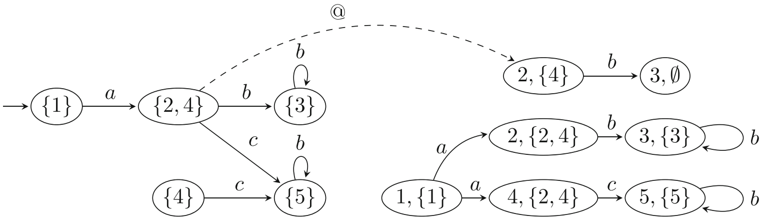


Fig. 11 The relevant part of the observer of G_3 (left), the corresponding part of the automaton C (right), and the @-transition (dashed) added by the algorithm

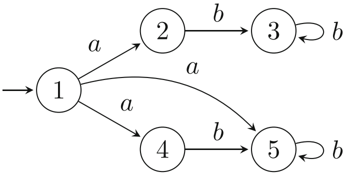
contradiction, assume that a state of the form $(q, \emptyset)$ is reachable in $\mathcal{C}$ from a state of Y. Then, there must be a string s such that $P(s)$ reaches a state X in the observer of G such that $X_S = X \cap Q_S$ contains a state z, $X \cap Q_{NS} = Z \neq \emptyset$, there is a transition under @ from X to the pair (z, Z) of $\mathcal{C}$, and the NFA $\mathcal{C}$ reaches state $(q, \emptyset)$ from (z, Z) under a string w. In particular, there must be a string $t \in P^{-1}(w)$ that moves G from state z to state q. But then $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$, and $\delta(\delta(I, P^{-1}P(s)) \cap Q_{NS}, P^{-1}(w)) = \emptyset$, which means that G is not infinite-step opaque – a contradiction. □

Since our algorithm constructs and searches the NFA $\mathcal{C}$ that has $O(n2^n)$ states and $O(m\ell 2^n)$ transitions, the overall time complexity of our algorithm is $O((n + m\ell)2^n)$.

We now illustrate the procedure in the following example.

Example 5 We consider system G_3 of Example 3 as depicted in Fig. 8 with all the events a, b, c observable, the set of secret states $Q_S = \{2\}$, and the set of non-secret states $Q_{NS} = \{4\}$. Then G_3 is current-state opaque, but not infinite-step opaque. To show that G_3 is not infinite-step opaque, our algorithm works as follows. First, notice that $P(G_3)$ coincides with G_3, since there are no unobservable transitions in G_3. A relevant part of the observer of G_3 is depicted in Fig. 11 (left), and a relevant part of the automaton C, i.e., of the product of $P(G_3)$ with the observer of G_3, is depicted in Fig. 11 (right). The only reachable state of the observer that has a nonempty intersection with $Q_S = \{2\}$ is state $X = \{2, 4\}$, resulting in $X_S = \{2\}$ and $X_{NS} = \{4\}$. The algorithm then creates an @-transition from state $X = \{2, 4\}$ of the observer to state $(2, \{4\})$ of the product automaton C (the dashed transition in Fig. 11). Since state $(3, \emptyset)$ is reachable from state $(2, \{4\})$ in C, system G_3 is not infinite-step opaque; indeed, observing ab in G_3, the intruder knows for sure that the system was in a secret state.

Fig. 12 Projected automaton $P(\tilde{G}_3)$



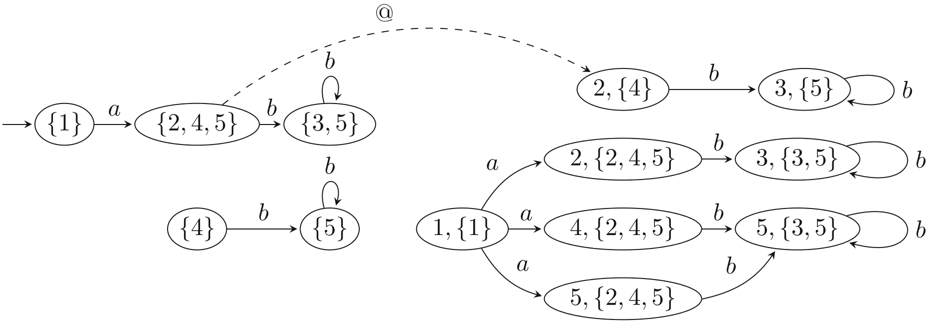


Fig. 13 The relevant part of the observer of $\tilde{G_3}$ (left), the corresponding part of the automaton C (right), and the @-transition (dashed) added by the algorithm

On the other hand, we now assume that c is unobservable. To avoid confusion, we denote G_3 with a and b observable, c unobservable, the set of secret states $Q_S = \{2\}$, and the set of non-secret states $Q_{NS} = \{4\}$ as $\tilde{G_3}$. Then $\tilde{G_3}$ is infinite-step opaque, and our algorithm works as follows. First, we construct $P(\tilde{G_3})$ as shown in Fig. 12. Relevant parts of the observer of $\tilde{G_3}$ and of the product of $P(\tilde{G_3})$ with the observer, automaton C, is depicted in Fig. 13.

The only reachable state of the observer of $\tilde{G_3}$ with a nonempty intersection with $Q_S = \{2\}$ is state $X = \{2, 4, 5\}$, resulting in $X_S = \{2\}$ and $X_{NS} = \{4\}$. The algorithm creates an @-transition from state $X = \{2, 4, 5\}$ of the observer to state (2, {4}) of the product automaton C (the dashed transition in Fig. 13). Since no state of the form $(q, \emptyset)$ is reachable from state (2, {4}), $\tilde{G_3}$ is infinite-step opaque.

4.3 Transformations between CSO and K-SO

In this section, we describe the transformations between current-state opacity and K-step opacity. To the best of our knowledge, no such transformations have been considered in the literature so far.

4.3.1 Transforming CSO to K-SO

The transformation from current state opacity to K-step opacity is analogous to the transformation from current state opacity to infinite-step opacity of Section 4.2.1. Intuitively, the modification is that we need to make only K observable steps from any non-secret state instead of infinitely many such steps.

The problem of deciding current-state opacity consists of a DES $G_{CSO} = (Q, \Sigma, \delta, I)$, a projection $P\colon \Sigma^* \to \Sigma_o^*$, a set of secret states $Q_S \subseteq Q$, and a set of non-secret states $Q_{NS} \subseteq Q$.

For a given $K \in \mathbb{N}$, from G_{CSO}, we construct a DES $G_{K\text{-}SO}$= $(Q \cup Q^\star, \Sigma \cup \{u\}, \delta', I)$, where u is a new unobservable event, by adding $K + 1$ new states $Q^\star = \{q_0^\star, \ldots, q_K^\star\}$

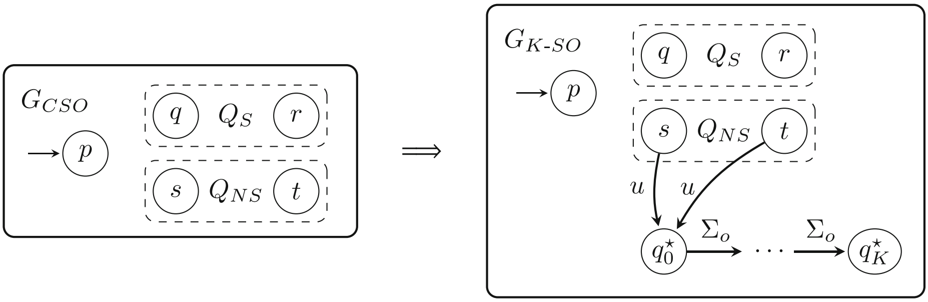


Fig. 14 Transforming CSO to K-SO

that are neither secret nor non-secret, and by defining δ' as follows, see Fig. 14 for an illustration:

1. $\delta' = \delta$, that is, δ' is initialized as δ and further extended as follows;
2. for every state $q \in Q_{NS}$, we add the transition $(q, u, q_0^\star)$ to δ';
3. for $i = 0, \dots, K-1$ and every $a \in \Sigma_o$, we add the transition $(q_i^\star, a, q_{i+1}^\star)$ to δ'.

We extend the projection P to the projection $P' \colon (\Sigma \cup \{u\})^* \to \Sigma_o^*$. The sets Q_S and Q_{NS} remain unchanged.

Theorem 7 *The DES G_{CSO} is current-state opaque with respect to Q_S, Q_{NS}, and P if and only if the DES $G_{K\text{-}SO}$ is K-step opaque with respect to Q_S, Q_{NS}, P', and K.*

Proof Assume first that G_{CSO} is not current-state opaque. Since the new states $q_0^\star, \dots, q_K^\star$ are neither secret nor non-secret, $G_{K\text{-}SO}$ is not current-state opaque either, and hence $G_{K\text{-}SO}$ is not K-step opaque.

On the other hand, assume that G_{CSO} is current-state opaque. Since the new states $q_0^\star, \dots, q_K^\star$ are neither secret nor non-secret, $G_{K\text{-}SO}$ is current-state opaque as well. Let $st \in L(G_{K\text{-}SO})$ be such that $|P(t)| \leq K$ and $\delta'(\delta'(I, s) \cap Q_S, t) \neq \emptyset$. Then, since $G_{K\text{-}SO}$ is current-state opaque, there is $s' \in P^{-1}P(s)$ such that $\delta'(I, s') \cap Q_{NS} \neq \emptyset$. By construction, we can extend s' by the string $uP(t)$ using the transitions through the new states $q_0^\star, \dots, q_K^\star$, that is, $\delta'(\delta'(I, s') \cap Q_{NS}, uP(t)) \neq \emptyset$, and hence $G_{K\text{-}SO}$ is K-step opaque. □

We now illustrate the construction.

Example 6 Let G_2 over $\Sigma = \{a, b, c\}$ depicted in Fig. 15 (left) be the instance of the CSO problem from Example 2 with the set of secret states $Q_S = \{2\}$ and the set of non-secret states $Q_{NS} = \{5\}$. Our transformation of CSO to K-SO then results in the DES G_2'' depicted in Fig. 15 (right) with $K = 2$, a new unobservable event u, and three new states $q_0^\star$, $q_1^\star$, and $q_2^\star$. We again distinguish two cases depending on whether event c is observable or not.

If c is unobservable, G_2 is current-state opaque as shown in Example 2. The reader can see that G_2'' is then 2-step opaque, because the only possible extensions of the string a from the secret state 2 are of the form b^k, for $k \in \mathbb{N}$, and for those extensions where $k \leq 2$, there is an extension ub^k of the string ac from the non-secret state 5 such that $P(ab^k) = P(acub^k)$.

If c is observable, then G_2 is not current-state opaque as shown in Example 2. Consequently, the reader can verify that G_2'' is not current-state opaque, and hence neither 2-step opaque.

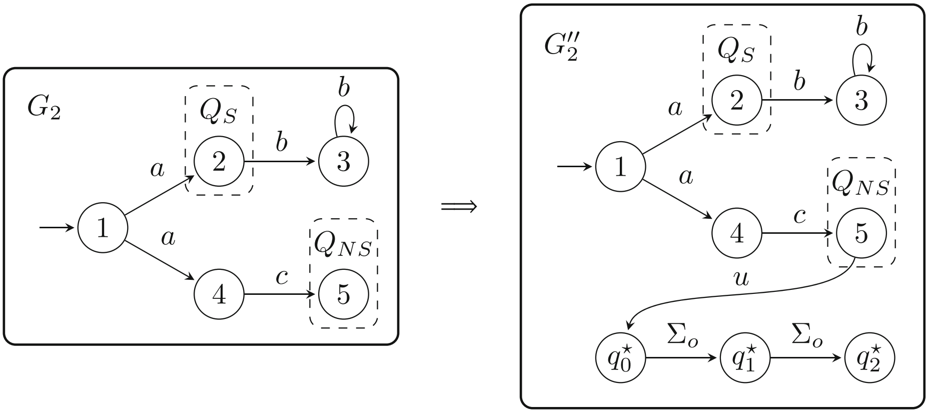


Fig. 15 An example of the transformation of the CSO problem (left) to the K-SO problem (right)

4.3.2 Transforming K-SO to CSO

Transforming K-step opacity to current-state opacity is again similar to the transformation of infinite-step opacity to current-state opacity. Again, we only need to check K subsequent steps instead of all the subsequent steps. The problem of deciding K-step opacity consists of a DES $G_{\text{K-SO}} = (Q, \Sigma, \delta, I)$, a projection $P : \Sigma^* \to \Sigma_o^*$, a set of secret states $Q_S \subseteq Q$, and a set of non-secret states $Q_{NS} \subseteq Q$. From $G_{\text{K-SO}}$, we construct a DES G_{CSO} in the following two steps:

1. We construct a DES G_{CSO} such that G_{CSO} is current-state opaque if and only if $G_{\text{K-SO}}$ is K-step opaque. In this step of the construction, G_{CSO} has one observable event more than $G_{\text{K-SO}}$.
2. To reduce the number of observable events by one, we apply Lemma 2. Consequently, the resulting DES has the same number of observable events as $G_{\text{K-SO}}$, if $G_{\text{K-SO}}$ has at least two observable events, is deterministic if and only if G_{CSO} is, and is current-state opaque if and only if G_{CSO} is.

We now describe the construction of $G_{CSO} = (Q \cup Q^+ \cup Q^- \cup Q^\star, \Sigma \cup \{u, @\}, \delta', I)$, where $Q^+ = \{q^+ \mid q \in Q\}$, $Q^- = \{q^- \mid q \in Q\}$, $Q^\star = \{q_0^\star, \ldots, q_{K+1}^\star\}$, @ is a new observable event, and u is a new unobservable event. To this end, we first make two disjoint copies of $G_{\text{K-SO}}$, denoted by G^+ and G^-, where the set of states of G^+ is denoted by Q^+ and the set of states of G^- is denoted by Q^-. The DES G_{CSO} is now taken as the disjoint union of the automata $G_{\text{K-SO}}$, G^+, and G^-, see Fig. 16 for an illustration. We now add K+2 new states $q_0^\star, \ldots, q_{K+1}^\star$ to G_{CSO} and the following transitions. For every state $q \in Q_S$, we add the transition $(q, @, q^+)$, for every state $q \in Q_{NS}$, we add the transition $(q, @, q^-)$, for every $q^- \in Q^-$, we add the transition $(q^-, u, q_0^\star)$, for every $a \in \Sigma_o$ and $i = 0, \ldots, K$, we add the transition $(q_i^\star, a, q_{i+1}^\star)$, and, finally, we add the self-loop $(q_{K+1}^\star, a, q_{K+1}^\star)$ for every $a \in \Sigma_o$. The set of secret states of G_{CSO} is the $Q_S' = Q^+$ and the set of non-secret states of G_{CSO} is the set $Q_{NS}' = \{q_0^\star, q_{K+1}^\star\}$. We extend projection P to $P' : (\Sigma \cup \{@, u\})^* \to (\Sigma_o \cup \{@\})^*$.

Notice that G_{CSO} is deterministic if and only if $G_{\text{K-SO}}$ is, and that logarithmic space is sufficient for the construction of G_{CSO}. However, as already pointed out, the construction

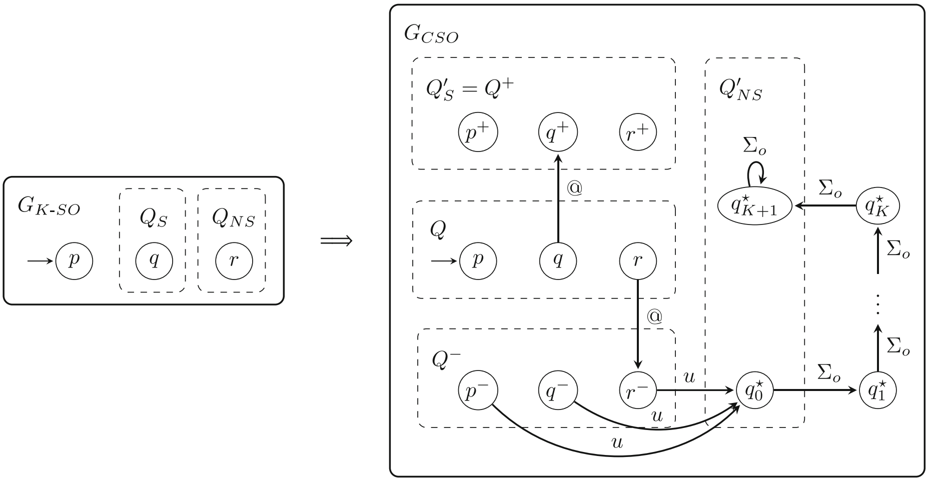


Fig. 16 Transforming K-SO to CSO

does not preserve the number of observable events, which requires the second step of the construction using Lemma 2.

We now show that $G_{\text{K-SO}}$ is K-step opaque if and only if G_{CSO} is current-state opaque.

Theorem 8 *The DES $G_{K\text{-}SO}$ is K-step opaque with respect to Q_S, Q_{NS}, and P if and only if the DES G_{CSO} is current-state opaque with respect to Q'_S, Q'_{NS}, and $P' \colon (\Sigma \cup \{@, u\})^* \to (\Sigma_o \cup \{@\})^*$.*

Proof Assume that $G_{\text{K-SO}}$ is K-step opaque. We show that G_{CSO} is current-state opaque. To this end, consider a string w such that $\delta'(I, w) \cap Q'_S \neq \emptyset$. We want to show that there exists $w' \in P'^{-1}P'(w)$ such that $\delta'(I, w') \cap Q'_{NS} \neq \emptyset$. However, since $Q'_S = Q^+$, w is of the form $w_1@w_2$ and, by the construction, $\delta(I, w_1)$ contains a secret state of $G_{\text{K-SO}}$. Since G is K-step opaque, there exists a string $w'_1 \in P^{-1}P(w_1)$ such that $\delta(I, w'_1) \cap Q_{NS} \neq \emptyset$. Then, because w_2 can be read in the copy of $G_{\text{K-SO}}$ from a state q^+ for a state $q \in \delta(I, w_1) \cap Q_S$, we further have that $\delta(\delta(I, w_1) \cap Q_S, w_2) \neq \emptyset$. If $|P(w_2)| \leq K$, then K-step opacity of $G_{\text{K-SO}}$ implies that there exists a string $w''_1 w''_2 \in L(G_{\text{K-SO}})$ such that $P(w''_1) = P(w_1)$, $P(w''_2) = P(w_2)$, and $\delta(\delta(I, w''_1) \cap Q_{NS}, w''_2) \neq \emptyset$. By construction, $q^\star_0 \in \delta'(\delta'(I, w''_1@) \cap Q_{NS}, w''_2 u)$, and hence G_{CSO} is current-state opaque. If $|P(w_2)| > K$, then $q^\star_{K+1} \in \delta'(\delta'(I, w''_1@) \cap Q_{NS}, uP(w''_2))$, and hence G_{CSO} is current-state opaque.

On the other hand, assume that $G_{\text{K-SO}}$ is not K-step opaque, that is, there exists a string $st \in L(G_{\text{K-SO}})$ such that $|P(t)| \leq K$, $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$ and, for every $s' \in P^{-1}P(s)$ and $t' \in P^{-1}P(t)$, $\delta(\delta(I, s') \cap Q_{NS}, t') = \emptyset$. But then, for $s@t \in L(G_{CSO})$, we have that $\delta'(\delta'(I, s@) \cap Q'_S, t) \cap Q'_S \neq \emptyset$ and, for every $s'@t' \in L(G_{CSO})$ such that $P'(s@t) = P'(s'@t')$, we have two cases:

(i) If $\delta(I, s') \cap Q_{NS} = \emptyset$, then $\delta'(I, s'@t') \cap Q'_{NS} = \delta'(\delta'(I, s'@) \cap Q^-, t') = \delta'(\emptyset, t') = \emptyset$, which shows that G_{CSO} is not current-state opaque.

(ii) If $\delta(I, s') \cap Q_{NS} \neq \emptyset$, then $\delta'(I, s'@t') \cap Q'_{NS} = \delta'(\delta'(I, s'@) \cap Q^-, t') = \emptyset$, because inserting u to any strict prefix of t' may reach $q^\star_0$ but has to leave it when the rest of t'

is read, and the rest (neither $P(t')$) is not long enough to reach state $q^{\star}_{K+1}$. Therefore, G_{CSO} is not current-state opaque.

□

We now illustrate the construction.

Example 7 Let G_3 over $\Sigma = \{a, b, c\}$ depicted in Fig. 17 (left) be the instance of the K-SO problem from Example 3 with $K = 2$, the set of secret states $Q_S = \{2\}$, and the set of non-secret states $Q_{NS} = \{4\}$. Our transformation of K-SO to CSO then results in the DES G''_3 depicted in Fig. 17 (right) with a new observable event @, a new unobservable event u, the set of secret states Q'_S, and the set of non-secret states Q'_{NS}. We consider two cases based on the observability status of event c.

If c is unobservable, then G_3 is 2-step opaque, because it is infinite-step opaque as shown in Example 3. The reader can further see that G''_3 is current-state opaque, because it can enter a secret state only after generating a string of the form $a@b^k$, for $k \in \mathbb{N}$, in which case we have that $\delta'(1, P^{-1}(a@)) = \{2^+, 4^-, 5^-, q^{\star}_0\}$ and $\delta'(1, P^{-1}(a@b^k)) = \{3^+, 5^-, q^{\star}_0, \ldots, q^{\star}_i\}$ for $k \geq 1$, where $i = \min\{k, 3\}$.

If c is observable, then G_3 is not 2-step opaque, because after generating string ab, the intruder can deduce that the system was in the secret state 2. Similarly, after generating string $a@b$, system G''_3 ends up in the only state 3^+, which is a secret state, and hence G''_3 is not current-state opaque.

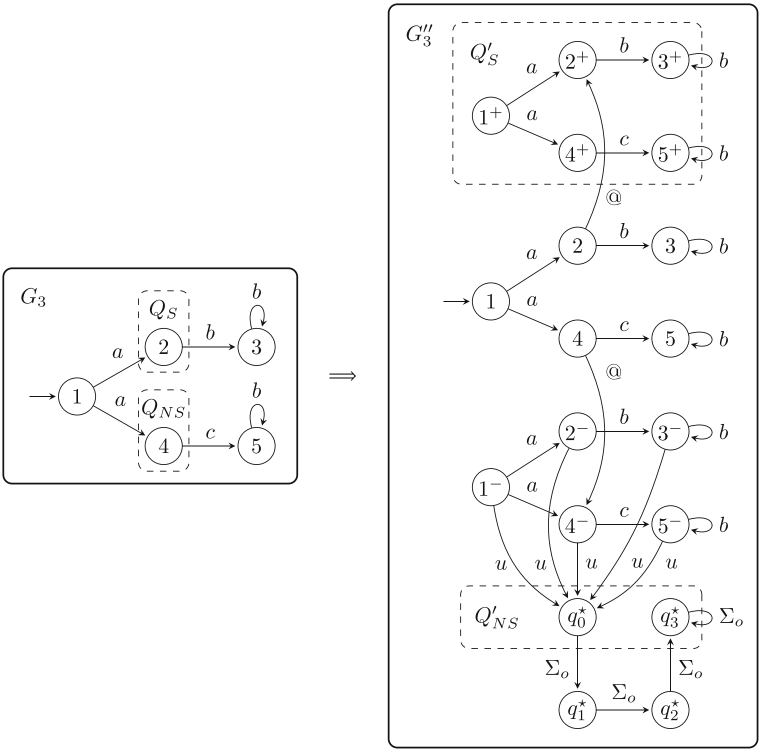


Fig. 17 An example of the transformation of the K-SO problem (left) to the CSO problem (right)

4.3.3 The case of a single observable event

To preserve the number of observable events, our transformation of K-step opacity to current state opacity relies on Lemma 2. This lemma requires at least two observable events in $G_{\text{K-SO}}$, and hence it is not applicable to systems with a single observable event. For these systems, we provide a different transformation that requires to add at most a quadratic number of new states.

The problem of deciding K-step opacity for systems with a single observable event consists of a DES $G_{\text{K-SO}} = (Q, \Sigma, \delta, I)$ with $\Sigma_o = \{a\}$, a set of secret states $Q_S \subseteq Q$, a set of non-secret states $Q_{NS} \subseteq Q$, and a projection $P \colon \Sigma^* \to \{a\}^*$. We denote the number of states of $G_{\text{K-SO}}$ by n, and define a function $\varphi \colon Q \to \{0, \ldots, K\}$ that assigns, to every state q, the maximal number $k \in \{0, \ldots, K\}$ of observable steps that are possible from state q; formally, $\varphi(q) = \max\left\{k \in \{0, \ldots, K\} \mid \delta(q, P^{-1}(a^k)) \neq \emptyset\right\}$. Notice that if $K > n - 1$, then a system with a single observable event is K-step opaque if and only if it is infinite-step opaque. Therefore, we may consider only $K \leq n - 1$.

From $G_{\text{K-SO}}$, we construct a DES $G_{CSO} = (Q', \Sigma, \delta', I)$ as illustrated in Fig. 18, where δ' is initialized as δ and modified as follows. For every state $p \in Q$ with $\varphi(p) > 0$, we add K new states $p_1, \ldots, p_K$ to Q' and K new transitions (p, a, p_1) and (p_i, a, p_{i+1}), for $i = 1, \ldots, K - 1$, to δ'. Finally, we replace every transition (p, a, r) in δ' by the transition (p_K, a, r). Notice that the transformation requires to add at most n^2 states, and hence it can be done in polynomial time. Let $Q'_S = Q_S$ and $Q'_{NS} = Q_{NS}$. For every state $p \in Q_S$ with $\varphi(p) = k > 0$, we add the corresponding states $p_1, \ldots, p_k$ to Q'_S and, for every $p \in Q_{NS}$ with $\varphi(p) = k > 0$, we add $p_1, \ldots, p_k$ to Q'_{NS}.

Notice that the transformation can be done in polynomial time, preserves the number of observable events, and determinism. However, whether the transformation can be done in logarithmic space is open.

We show that $G_{\text{K-SO}}$ is K-step opaque if and only if G_{CSO} is current-state opaque.

Theorem 9 *The DES $G_{\text{K-SO}}$ with a single observable event is K-step opaque with respect to Q_S, Q_{NS}, and P if and only if the DES G_{CSO} is current-state opaque with respect to Q'_S, Q'_{NS}, and P.*

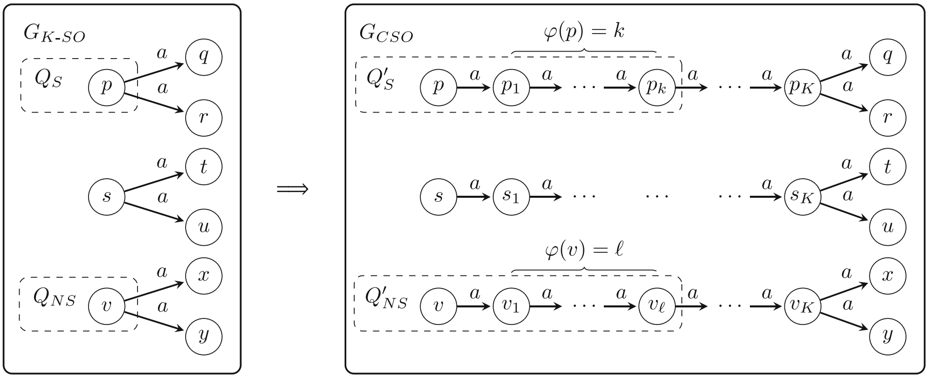


Fig. 18 Transforming K-SO to CSO for systems with a single observable event

Proof Assume that $G_{\text{K-SO}}$ is not K-step opaque, that is, there is $st \in L(G_{\text{K-SO}})$ with $|P(t)| \le K$ such that $\delta(\delta(I,s) \cap Q_S, t) \neq \emptyset$ and $\delta(\delta(I, P^{-1}P(s)) \cap Q_{NS}, P^{-1}P(t)) = \emptyset$. Let $f\colon \Sigma^* \to \Sigma^*$ be a morphism such that $f(a) = a^{K+1}$ and $f(b) = b$, for $a \neq b \in \Sigma$. Then, by construction, $\delta(I,s) = \delta'(I, f(s))$, and hence $\delta'(I, f(s)) \cap Q'_S \neq \emptyset$. If $\delta(I, P^{-1}P(s)) \cap Q_{NS} = \emptyset$, then $\delta'(I, f(P^{-1}P(s))) \cap Q'_{NS} = \emptyset$ because $\delta(I,s') = \delta'(I, f(s'))$ for any $s' \in P^{-1}P(s)$, and G_{CSO} is not current-state opaque. Otherwise, we denote by $q_s \in \delta(I,s) \cap Q_S$ and $q_{ns} \in \delta(I, P^{-1}P(s)) \cap Q_{NS}$ the states with maximal $\varphi(q_s)$ and $\varphi(q_{ns})$. Since $G_{\text{K-SO}}$ is not K-step opaque, $\varphi(q_s) > \varphi(q_{ns})$. Then, in G_{CSO}, q_s has exactly one outgoing observable transition and is followed by $\varphi(q_s) = k$ secret states, while q_{ns} is followed by $\varphi(q_{ns}) < k$ non-secret states. Therefore, $\delta'(I, f(s)a^k) \cap Q'_S \neq \emptyset$ and $\delta'(I, f(s')a^k) \cap Q'_{NS} = \emptyset$ for any $s' \in P^{-1}P(s)$, and hence G_{CSO} is not current-state opaque.

On the other hand, assume that $G_{\text{K-SO}}$ is K-step opaque, and that $\delta'(I,w) \cap Q'_S \neq \emptyset$. We show that $\delta'(I, P^{-1}P(w)) \cap Q_{NS} \neq \emptyset$. Consider a state $q_s \in \delta'(I,w) \cap Q'_S$ and a path π in G_{CSO} leading to q_s under w. Denote by p the last state of π that corresponds to a state of $G_{\text{K-SO}}$; that is, p is not a new state added by the construction of G_{CSO}. Since $q_s \in Q'_S$, we have, by construction, that $p \in Q_S$. Then the choice of p partitions $w = uv$, where u, read along the path π, leads to state p, and $v = a^\ell$ is a suffix of length $\ell \le K$. Let u' be a string such that $f(u') = u$. Then $p \in \delta(I,u') \cap Q_S$. Since $\varphi(p) \ge \ell$, there exists t such that $P(t) = a^\ell$ and $\delta(\delta(I,u') \cap Q_S, t) \neq \emptyset$ in $G_{\text{K-SO}}$. Then K-step opacity of $G_{\text{K-SO}}$ implies that there exists u'' and t' such that $P(u') = P(u'')$, $P(t) = P(t')$, and $\delta(\delta(I,u'') \cap Q_{NS}, t') \neq \emptyset$. In particular, there is a state $q_{ns} \in \delta(I,u'') \cap Q_{NS}$ with $\varphi(q_{ns}) \ge \ell$, and $\delta'(I, f(u'')) \cap Q'_{NS} \neq \emptyset$. Therefore, $\delta'(I, f(u'')a^\ell) \cap Q'_{NS} \neq \emptyset$ and $P(f(u'')a^\ell) = P(uv) = P(w)$, which completes the proof. □

We now illustrate the construction.

Example 8 Let G_4 over $\Sigma = \{a, u\}$ depicted in Fig. 19 (left) be the instance of the K-SO problem from Example 4 with $K = 2$, a single observable event $\Sigma_o = \{a\}$, the set of secret states $Q_S = \{1\}$, and the set of non-secret states $Q_{NS} = \{3\}$. Then, $\varphi(1) = \varphi(3) = 2$, and our transformation of K-SO to CSO results in the DES G''_4 depicted in Fig. 19 (right) with the set of secret states Q'_S and the set of non-secret states Q'_{NS}. Analogously to Example 4, we consider two cases based on the presence of the unobservable transition $(1, u, 3)$ in G_4.

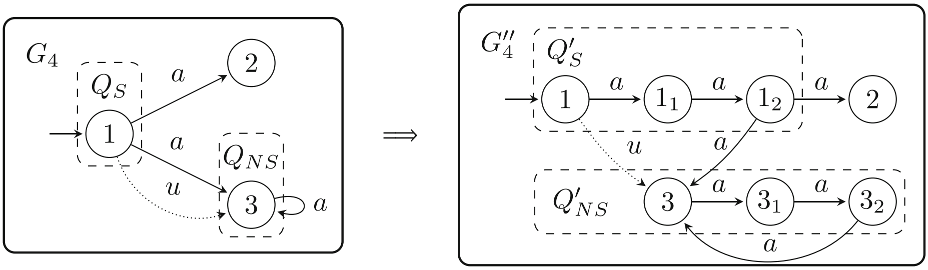


Fig. 19 An example of the transformation of the K-SO problem with a single observable event (left) to the CSO problem (right)

If the transition $(1, u, 3)$ exists in G_4, then G_4 is 2-step opaque, since it is infinite-step opaque as shown in Example 4. The reader can see that G_4'' is current-state opaque, because a secret state is reachable only under a string of the form a^k for $k \in \{0, 1, 2\}$, and for any such string there is an indistinguishable string ua^k reaching a non-secret state.

If the transition $(1, u, 3)$ does not exist in G_4, then G_4 is not 2-step opaque, because it is neither current-state opaque and, obviously, neither G_4'' is current-state opaque.

4.3.4 Improving the algorithmic complexity of deciding K-step opacity

Let $G = (Q, \Sigma, \delta, I, F)$ be a DES. We design an algorithm deciding K-step opacity in time $O((K + 1)2^n(n + \ell^2 m))$, where $\ell = |\Sigma_o|$ is the number of observable events, n is the number of states of G, and m is the number of transitions of $P(G)$, $m \leq \ell n^2$.

To decide whether G is K-step opaque with respect to $Q_S, Q_{NS} \subseteq Q$, and $P : \Sigma^* \to \Sigma_o^*$, we proceed as follows:

1. We compute the observer $\mathcal{G}^{obs}$ of G in time $O(\ell 2^n)$;
2. We compute the projected automaton $P(G)$ of G in polynomial time $O(m + n)$;
3. We compute a DFA $\mathcal{D}$ accepting the language Σ_o^K; then $\mathcal{D}$ has $K + 1$ states and is constructed in time $O(\ell(K + 1))$;
4. We compute the product automaton $\mathcal{C} = P(G) \times \mathcal{G}^{obs}$ in time $O((m + n) \cdot \ell 2^n)$; – states of $\mathcal{C}$ are of the form $Q \times 2^Q$;
5. For every reachable state X of $\mathcal{G}^{obs}$, we compute $X_S = X \cap Q_S$ and $X_{NS} = X \cap Q_{NS}$;
   (a) If $X_S \neq \emptyset$ and $X_{NS} = \emptyset$, then G is not K-step opaque;
   (b) Otherwise, for every state $x \in X_S$, we add a transition from X under @ to state (x, X_{NS}) of $\mathcal{C}$, and we add the state (x, X_{NS}) to set Y;
6. We set Y to be the set of initial states of $\mathcal{C}$, and compute $\mathcal{G} = \mathcal{C} \times \mathcal{D}$;
   (a) If $\mathcal{G}$ contains a reachable state of the form $(q, \emptyset, d)$, then G is not K-step opaque; otherwise, G is K-step opaque.

Informally, we make use of the algorithm designed for deciding infinite-step opacity of Section 4.2.4 with the modification that we take an intersection of $\mathcal{C}$ with the automaton recognizing Σ_o^K. This modification ensures that any computation of $\mathcal{C}$ ends after K steps, and hence we check at most K subsequent steps.

Lemma 4 *The DES G is K-step opaque if and only if G is current-state opaque and no state of the form $(q, \emptyset, d)$ is reachable in $\mathcal{G}$.*

Proof The algorithm works as that deciding infinite-step opacity. The only modification is that we intersect $\mathcal{C}$ with $\mathcal{D}$, recognizing Σ_o^K. This modification ensures that the algorithm checking infinite-step opacity is blocked after K subsequent steps, and hence it decides K-step opacity. □

Since our algorithm constructs and searches the NFA $\mathcal{G}$ with $O((K + 1)n2^n)$ states and $O((K + 1)\ell m 2^n \ell)$ transitions, the time complexity of our algorithm is $O((K + 1)2^n(n + \ell^2 m))$.

5 Conclusions

We studied the transformations among the notions of language-based opacity, current-state opacity, initial-state opacity, initial-and-final-state opacity, K-step opacity, and infinite-step opacity. In particular, we provided a general transformation from language-based opacity to initial-state opacity, and constructed transformations between infinite-step opacity and current-state opacity, and between K-step opacity and current-state opacity. Together with the transformations of Wu and Lafortune (2013), we have a complete list of transformations between the discussed notions of opacity. The transformations are computable in polynomial time, preserve the number of observable events, and determinism. We further applied the transformations to improve the algorithmic complexity of deciding language-based opacity, infinite-step opacity, and K-step opacity, and to obtain the precise computational complexity of deciding the discussed notions of opacity.

Acknowledgements We gratefully acknowledge suggestions and comments of the anonymous referees.

References

Alur R, Černý P, Zdancewic S (2006) Preserving secrecy under refinement. In: International colloquium on automata, languages and programming (ICALP), pp 107–118

Arora S, Barak B (2009) Computational complexity – A modern approach. Cambridge University Press

Asveld PRJ, Nijholt A (2000) The inclusion problem for some subclasses of context-free languages. Theor Comput Sci 230(1-2):247–256

Badouel E, Bednarczyk M, Borzyszkowski A, Caillaud B, Darondeau P (2007) Concurrent secrets. Discret Event Dyn Syst 17(4):425–446

Balun J, Masopust T (2020) On opacity verification for discrete-event systems. In: IFAC World congress, pp 2105–2110

Bryans JW, Koutny M, Mazarė L, Ryan PYA (2008) Opacity generalised to transition systems. Int J Inf Secur 7(6):421–435

Bryans JW, Koutny M, Ryan PYA (2005) Modelling opacity using Petri nets. Electron Notes Theor Comput Sci 121:101–115

Cassandras CG, Lafortune S (2008) Introduction to Discrete Event Systems, 2nd edn. Springer

Cassez F, Dubreil J, Marchand H (2012) Synthesis of opaque systems with static and dynamic masks. Formal Methods Syst Des 40(1):88–115

Domaratzki M, Salomaa K (2007) Transition complexity of language operations. Theor Comput Sci 387(2):147–154

Dubreil J, Darondeau P, Marchand H (2008) Opacity enforcing control synthesis. In: Workshop on discrete event systems (WODES), pp 28–35

Focardi R, Gorrieri R (1994) A taxonomy of trace-based security properties for ccs. In: Computer security foundations workshop VII, pp 126–136

Hadj-Alouane NB, Lafrance S, Lin F, Mullins J, Yeddes MM (2005) On the verification of intransitive noninterference in mulitlevel security. IEEE Trans Syst Man Cybern Part B 35(5):948–958

Holzer M, Kutrib M (2011) Descriptional and computational complexity of finite automata—A survey. Inf Comput 209(3):456–470

Hopcroft JE, Ullman JD (1979) Introduction to automata theory, languages and computation. Addison-Wesley

Immerman N (1988) Nondeterministic space is closed under complementation. SIAM J Comput 17:935–938

Jacob R, Lesage J, Faure J (2016) Overview of discrete event systems opacity: models, validation, and quantification. Annu Rev Control 41:135–146

Jiråskovå G, Masopust T (2012) On a structural property in the state complexity of projected regular languages. Theor Comput Sci 449:93–105

Jones ND (1975) Space-bounded reducibility among combinatorial problems. J Comput Syst Sci 11(1):68–85

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

Mazarė L. (2004) Decidability of opacity with non-atomic keys. In: Formal aspects in security and trust, pp 71–84

Saboori A (2011) Verification and enforcement of state-based notions of opacity in discrete event systems. Ph.D. thesis University of Illinois at Urbana-Champaign

Saboori A, Hadjicostis CN (2007) Notions of security and opacity in discrete event systems. In: Conference on decision and control (CDC), pp 5056–5061

Saboori A, Hadjicostis CN (2008) Opacity-enforcing supervisory strategies for secure discrete event systems. In: Conference on decision and control. IEEE

Saboori A, Hadjicostis CN (2011) Verification of K-step opacity and analysis of its complexity. IEEE Trans Autom Sci Eng 8(3):549–559

Saboori A, Hadjicostis CN (2012) Verification of infinite-step opacity and complexity considerations. IEEE Trans Autom Control 57(5):1265–1269

Schneider S, Sidiropoulos A (1996) CSP and anonymity. In: European symposium on research in computer security (ESORICS), LNCS, vol 1146, pp 198–218

Stockmeyer LJ, Meyer AR (1973) Word problems requiring exponential time: Preliminary report. In: ACM Symposium on theory of computing (STOC), pp 1–9

Szelepcsėnyi R (1988) The method of forced enumeration for nondeterministic automata. Acta Inf 26:279–284

Wong K (1998) On the complexity of projections of discrete-event systems. In: Workshop on discrete event systems (WODES), pp 201–206

Wu YC, Lafortune S (2013) Comparative analysis of related notions of opacity in centralized and coordinated architectures. Discrete Event Dyn Syst 23(3):307–339

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

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Jiří Balun received B.Sc. from Palacky University, Olomouc, Czechia in 2016, and the M.Sc. from Palacky University, Olomouc, Czechia in 2019, both in computer science. He is currently a Ph.D. candidate in the Computer Science program at the Palacky University, Olomouc. His research interests include verification and control of discrete-event systems.

Tomáš Masopust received M.Sc. in computer science from Masaryk University, Brno, Czechia in 2004, Ph.D. in computer science from Brno University of Technology, Brno, Czechia in 2007, and Res. Prof. in Informatics and Cybernetics from the Czech Academy of Sciences, Prague, Czechia in 2019. He worked at CWI Amsterdam, the Netherlands, in the Systems and Control Group, at the University of Bayreuth, Germany, in the Theoretical Computer Science Group, and at TU Dresden, Germany, in the Knowledge-Based Systems Group. His research interest includes verification and control of discrete-event systems and theoretical computer science. He is an associate editor of journals Kybernetika and JDEDS.