# K-Step Opacity in Discrete Event Systems: Verification, Complexity, and Relations

Jiří Balun and Tomáš Masopust

*Abstract*—Opacity is a property expressing whether a system may reveal its secret to a passive observer (an intruder) who knows the structure of the system but has a limited observation of its behavior. Several notions of opacity have been studied, including current-state opacity, K-step opacity, and infinite-step opacity. We study K-step opacity that generalizes both current-state opacity and infinite-step opacity, and asks whether the intruder cannot decide, at any time, whether or when the system was in a secret state during the last K observable steps. We design a new algorithm deciding K-step opacity the complexity of which is lower than that of existing algorithms and that does not depend on K. We then compare K-step opacity with other opacity notions and provide new transformations among the notions that do not use states that are neither secret nor non-secret (neutral states) and that are polynomial with respect to both the size of the system and the binary encoding of K.

*Index Terms*—K-Step Opacity, Discrete event systems, Verification, Complexity, Transformations

## I. INTRODUCTION

**P**ROPERTIES that guarantee to keep some information in a system secret include anonymity [1], noninterference [2], secrecy [3], security [4], and opacity [5]. In this paper, we are interested in opacity of systems modeled by finite automata.

Opacity is an information flow property asking if a system prevents an intruder from revealing the secret. The intruder is a passive observer that knows the structure of the system but has only limited observations of its behavior. Intuitively, the intruder estimates the behavior of the system, and the system is opaque if for every secret behavior, there is a non-secret behavior that looks the same to the intruder.

There are two common ways to model the secret: a set of secret states and a set of secret behaviors. In the former case, the opacity is referred to as state-based, introduced by Bryans et al. [6], [7] for systems modeled by Petri nets and transition systems, and later adapted to (stochastic) automata by Saboori and Hadjicostis [8]. In the latter case, the opacity is referred to as language-based, introduced by Badouel et al. [9] and Dubreil et al. [10]. For more details, see Jacob et al. [11].

Several opacity notions have been studied in the literature, including language-based opacity (LBO), initial-state opacity (ISO), initial-and-final-state opacity (IFO), current-state opacity (CSO), K-step opacity (K-SO), and infinite-step opacity (INSO). While initial-state opacity prevents the intruder from revealing, at any step of the computation, whether the system

J. Balun and T. Masopust are with the Department of Computer Science, Faculty of Science, Palacky University in Olomouc, Czechia. E-mails: jiri.balun01@upol.cz, tomas.masopust@upol.cz.

started in a secret state, current-state opacity prevents the intruder only from revealing whether the current state of the system is secret. The intruder may, however, realize in the future that the system was in a secret state at a former step of the computation. For example, if the intruder estimates that the system is in one of two 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 led to the introduction of K-step opacity [8], [12]. K-step opacity requires that the intruder cannot ascertain the secret in the current state and K subsequent observable steps. Two special cases for K = 0 and K = ∞ are known as current-state opacity and infinite-step opacity, respectively, though the notion of infinite-step opacity may be confusing for finite automata, because an automaton with $n$ states is infinite-step opaque if and only if it is $(2^n - 2)$-step opaque [13].

The complexity of known algorithms deciding K-step opacity depends on K. For example, the two-way observer of Yin and Lafortune [13] has complexity $O(\min\{n2^{2n}, n\ell^K 2^n\})$, including a minor correction by Lan et al. [14], where $n$ is the number of states of the automaton and $\ell$ is the number of observable events. Obviously, the complexity depends on K if $\ell^K < 2^n$. We recently designed an algorithm with complexity $O((K + 1)2^n(n + m\ell^2))$, where $m \leq \ell n^2$ is the number of transitions in the projected NFA, which is faster than the two-way observer if K is larger than $2^n - 2$ or polynomial in $n$ [15]. The reader can find more methods with their experimental comparisons in Wintenberg et al. [16].

In this paper, we further improve the complexity of deciding K-step opacity to $O((n+m)2^n)$, which does not depend on K. We then provide new transformations among K-step opacity, current-state opacity, and infinite-step opacity. These transformations have been studied by Balun and Masopust [15], who have shown that the notions are transformable to each other in polynomial time, and the results do not have more observable events and preserve determinism. However, the transformations from K-step opacity are polynomial only if K is small or considered as constant, whereas a large value of K makes the transformations infeasible. In addition, some of the transformations use *neutral states*—states that are neither secret nor non-secret.

Here we suggest new transformations that are polynomial in both the size of the system and a binary encoding (logarithm) of K, and that do not use neutral states. Wu and Lafortune [17] studied transformations among other notions of opacity. We refer the reader to Balun and Masopust [15] for an overview of these transformations and the complexity results.

## II. PRELIMINARIES

We assume that the reader is familiar with discrete-event systems [18]. For a set $S$, $|S|$ denotes the cardinality of $S$ and $2^S$ its power set. An alphabet $\Sigma$ is a finite nonempty set of events. A string over $\Sigma$ is a sequence of events; the empty string is denoted by $\varepsilon$. The set of all finite strings over $\Sigma$ is denoted by $\Sigma^*$. A language $L$ over $\Sigma$ is a subset of $\Sigma^*$. The set of 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 $u \in \Sigma^*$, $|u|$ is the length of $u$.

A *nondeterministic finite automaton* (NFA) over an alphabet $\Sigma$ 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. For a set $Q_0 \subseteq Q$, the set $L_m(\mathcal{G}, Q_0) = \{w \in \Sigma^* \mid \delta(Q_0, w) \cap F \neq \emptyset\}$ is the language marked by $\mathcal{G}$ from the states of $Q_0$, and $L(\mathcal{G}, Q_0) = \{w \in \Sigma^* \mid \delta(Q_0, w) \neq \emptyset\}$ is the language generated by $\mathcal{G}$ from $Q_0$. The languages *marked* and *generated* by $\mathcal{G}$ are $L_m(\mathcal{G}) = L_m(\mathcal{G}, I)$ and $L(\mathcal{G}) = L(\mathcal{G}, I)$, respectively. For $S \subseteq \Sigma^*$, we write $\delta(Q, S) = \cup_{s \in S} \delta(Q, s)$. The NFA $\mathcal{G}$ is *deterministic* (DFA) if $|I| = 1$ and $|\delta(q, a)| \leq 1$ for every $q \in Q$ and $a \in \Sigma$.

A *discrete-event system* (DES) $G$ over $\Sigma$ is an NFA over $\Sigma$ together with the partition of $\Sigma$ into $\Sigma_o$ and $\Sigma_{uo}$ of *observable* and *unobservable events*, respectively. If the marked states are irrelevant, we omit them and simply write $G = (Q, \Sigma, \delta, I)$.

State estimation is modeled by *projection* $P \colon \Sigma^* \to \Sigma_o^*$, which is a morphism defined by $P(a) = \varepsilon$ if $a \in \Sigma_{uo}$, and $P(a) = a$ if $a \in \Sigma_o$. The action of $P$ on a string $a_1 \cdots a_n$ is to erase unobservable events: $P(a_1 \cdots a_n) = P(a_1) \cdots P(a_n)$. The definition can be readily extended to languages.

Let $G$ be a DES over $\Sigma$ with projection $P \colon \Sigma^* \to \Sigma_o^*$. The *projected automaton* of $G$ is the NFA $P(G)$ obtained from $G$ by replacing every transition $(p, a, q)$ by $(p, P(a), q)$, and by eliminating the $\varepsilon$-transitions. In particular, if $\delta$ is the transition function of $G$, then the transition function $\gamma \colon Q \times \Sigma_o \to 2^Q$ of $P(G)$ is defined as $\gamma(q, a) = \delta(q, P^{-1}(a))$. Then, $P(G)$ is an NFA over $\Sigma_o$ with the same states as $G$ that recognizes the language $P(L_m(G))$ and that can be constructed in polynomial time [19]. The DFA constructed from $P(G)$ by the standard subset construction is called an *observer of $G$* [18], which has up to exponentially more states than $G$ [20], [21].

## III. K-STEP OPACITY AND ITS VERIFICATION

We denote the set of non-negative integers by $\mathbb{N}$. For $K \in \mathbb{N}_\infty = \mathbb{N} \cup \{\infty\}$, K-step opacity asks if the intruder cannot reveal the secret in the current and $K$ subsequent states.

**Definition 1.** *Given a DES $G = (Q, \Sigma, \delta, I)$ and $K \in \mathbb{N}_\infty$. System $G$ is K-step opaque (K-SO) w.r.t. secret states $Q_S$, non-secret states $Q_{NS}$, and $P \colon \Sigma^* \to \Sigma_o^*$ if for every string $st \in L(G)$ with $|P(t)| \leq K$ and $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$, there is $s't' \in L(G)$ such that $P(s) = P(s')$, $P(t) = P(t')$, and $\delta(\delta(I, s') \cap Q_{NS}, t') \neq \emptyset$.*

Two special cases of K-step opacity include 0-step opacity also known as *current-state opacity (CSO)*, and $\infty$-step opacity aka *infinite-step opacity (INSO)* [12], which, for a DES with $n$ states, coincides with $(2^n - 2)$-step opacity [13].

The complexity of existing algorithms verifying K-SO is exponential and depends on K. Exponential complexity seems unavoidable because the problem is PSPACE-complete [15]. We now design an algorithm verifying K-SO with complexity $O((n+m)2^n)$, where $n$ is the number of states of the automaton and $m$ is the number of transitions of the projected NFA, which improves the existing complexity and does not depend on K. Comparing the complexity with that of Wintenberg et al. [16], who neglect the number of transitions in the automata, our complexity can be stated as $O(n2^n)$, which is better than the results in Wintenberg et al. [16].

---

**Algorithm 1:** Verification of K-step opacity

**Input** : A DES $G = (Q, \Sigma, \delta, I)$, $Q_S, Q_{NS} \subseteq Q$, $\Sigma_o \subseteq \Sigma$, and $K \in \mathbb{N}_\infty$.

**Output:** `true` if and only if $G$ is K-SO w.r.t. $Q_S$, $Q_{NS}$, and $P \colon \Sigma^* \to \Sigma_o^*$

---

**1** Set $Y := \emptyset$

**2** Compute the observer $G^{obs}$ of $G$

**3** Compute the projected automaton $P(G)$ of $G$

**4** **for** *every reachable state $X$ of $G^{obs}$* **do**

**5**      **for** *every state $x \in X \cap Q_S$* **do**

**6**          add state $(x, X \cap Q_{NS})$ to set $Y$

**7** Compute the product automaton $\mathcal{C} = P(G) \times G^{obs}$ with the states of $Y$ as initial states

**8** Use BFS to mark states of $\mathcal{C}$ reachable from $Y$ in at most $K$ steps

**9** **if** $\mathcal{C}$ *contains a marked state of the form $(q, \emptyset)$* **then** **return** `false` **else return** `true`

---

Our algorithm is described as Algorithm 1. Intuitively, we compute the observer of $G$ (on demand also some of its non-reachable states), the projected NFA of $G$, and their product automaton $\mathcal{C}$. For every reachable state $X$ of the observer, we make the states $(x, X \cap Q_{NS})$, where $x$ is a secret state from $X$ and the second component is a set of all non-secret states from $X$, initial in $\mathcal{C}$. Then we use Breadth-First Search (BFS) [22] to search $\mathcal{C}$ and to mark all states of $\mathcal{C}$ that are reachable in at most K steps from an initial state. This is done as follows. First, we push all initial states of $\mathcal{C}$ to the queue, followed by pushing number 0 (in binary) to the queue. After processing the initial states, we remove 0 from and push 1 to the queue. At this point, the queue contains all states of $\mathcal{C}$ reachable from the initial states in one step, followed by number 1. The algorithm proceeds this way until it has either visited all states of $\mathcal{C}$ or the number stored in the queue is K. All and only visited states of $\mathcal{C}$ are marked. We show in Theorem 2 that $G$ is K-SO if and only if no state of the form $(\cdot, \emptyset)$ is marked in $\mathcal{C}$.

Before that, we illustrate Algorithm 1 by considering one-step opacity of the DES $G$ depicted in Figure 1 where all events are observable, $Q_S = \{2\}$, and $Q_{NS} = \{4\}$. A relevant part of the observer $G^{obs}$ is depicted in the same figure. Since $G$ has no unobservable events, $P(G) = G$. The only reachable state $X = \{2, 4\}$ of $G^{obs}$ intersecting $Q_S$ results in $Y = \{(2, \{4\})\}$. The marked part of $\mathcal{C}_1 = P(G) \times G^{obs}$ reachable from $Y$ in at most one step is depicted in Figure 2. Since
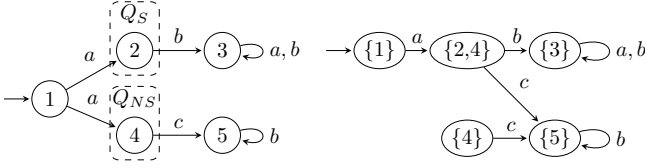
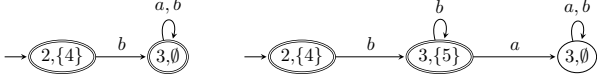Figure 1. A DES $G$ (left) and a relevant part of the observer $G^{obs}$ (right).



Figure 2. The reachable parts of $\mathcal{C}_1$ (left) and $\mathcal{C}_2$ (right).

state $(3, \emptyset)$ is marked in $\mathcal{C}_1$, $G$ is not one-step opaque; indeed, observing $ab$, the intruder reveals that $G$ was in a secret state.

We now assume that event $c$ is unobservable, and denote $G$ with $a, b$ observable, $c$ unobservable, $Q_S = \{2\}$, $Q_{NS} = \{4\}$ by $\tilde{G}$. The automaton $P(\tilde{G})$ and a relevant part of $\tilde{G}^{obs}$ are depicted in Figure 3. The only reachable state $X = \{2, 4, 5\}$ of $\tilde{G}^{obs}$ intersecting $Q_S$ results in $Y = \{(2, \{4\})\}$. The marked part of $\mathcal{C}_2 = P(\tilde{G}) \times \tilde{G}^{obs}$ is depicted in Figure 2. Since no state of the form $(\cdot, \emptyset)$ is marked in $\mathcal{C}_2$, $\tilde{G}$ is one-step opaque.

We now prove the correctness of our algorithm.

**Theorem 2.** *A DES $G$ is K-SO w.r.t. $Q_S$, $Q_{NS}$, and $P$ if and only if Algorithm 1 returns* true.

*Proof.* If $G = (Q, \Sigma, \delta, I)$ is not K-SO, then there exists $st \in L(G)$ such that $|P(t)| \leq K$, $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$, and $\delta(\delta(I, P^{-1}P(s)) \cap Q_{NS}, P^{-1}P(t)) = \emptyset$. We have two cases. (i) If $\delta(I, P^{-1}P(s)) \cap Q_{NS} = \emptyset$, then $G$ is not K-SO. The algorithm detects this situation by $X = \delta(I, P^{-1}P(s))$, since there is $q \in X \cap Q_S \neq \emptyset$ and $X \cap Q_{NS} = \emptyset$, which results in adding $(q, \emptyset)$ to $Y$ in line 6. (ii) If $\delta(I, P^{-1}P(s)) \cap Q_{NS} = Z \neq \emptyset$, then the pairs $(\delta(I, P^{-1}P(s)) \cap Q_S) \times \{Z\}$ are added to $Y$. Since $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$, there is $(z, Z) \in Y$ such that $P(t)$ leads the 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 $(q, \emptyset)$ is reachable in $\mathcal{C}$ from a state of $Y$ in at most $|P(t)| \leq K$ steps.

On the other hand, if $G$ is K-SO, we show that no state of the form $(q, \emptyset)$ is reachable in $\mathcal{C}$ from $Y$ in at most K steps. For the sake of contradiction, assume that a state $(q, \emptyset)$ is marked in $\mathcal{C}$. Then, there must be a string $s$ such that $\delta(I, P^{-1}P(s)) = X$ in $G$, that is, $P(s)$ reaches state $X$ in the observer of $G$, and there is $z \in X \cap Q_S$, $X \cap Q_{NS} = Z$, $(z, Z) \in Y$, and state $(q, \emptyset)$ is reached from state $(z, Z)$ in $\mathcal{C}$ by a string $w \in \Sigma_o^*$ of length at most K. In particular, there is $t \in P^{-1}(w)$ moving $G$ from state $z$ to state $q$. But then $q \in \delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$, and $\delta(\delta(I, P^{-1}P(s)) \cap Q_{NS}, P^{-1}(w)) = \delta(Z, P^{-1}(w)) = \emptyset$, which means that $G$ is not K-SO—a contradiction. $\square$
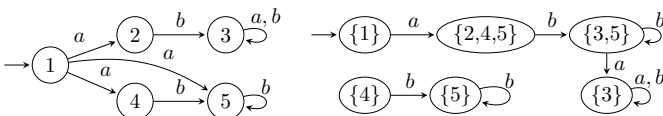


Figure 3. Projected automaton $P(\tilde{G})$ (left) and a relevant part of $\tilde{G}^{obs}$ (right).

Finally, we discuss the complexity of our algorithm.

**Theorem 3.** *The space and time complexity of Algorithm 1 is resp. $O(n2^n)$ and $O((n + m)2^n)$, where $n$ is the number of states of $G$ and $m$ is the number of transitions of $P(G)$. Further, $m \leq \ell n^2$, where $\ell$ is the number of observable events.*

*Proof.* Computing the observer and the projected NFA, lines 2 and 3, takes time $O(\ell 2^n)$ and $O(m + n)$, resp. The cycle on lines 4–6 takes $O(n2^n)$. Constructing $\mathcal{C}$, line 7, takes time $O(n2^n + m2^n)$, where $O(n2^n)$ is the number of states and $O(m2^n)$ is the number of transitions of $\mathcal{C}$. The BFS takes time linear in $\mathcal{C}$, and the condition of line 9 can be processed during the BFS. Since $m \geq \ell$, the proof is complete. $\square$

## IV. RELATION TO OTHER OPACITY NOTIONS

We now design polynomial-time transformations of K-SO to CSO, and vice versa. For the transformations of CSO to other opacity notions, we refer the reader to the literature [15], [17]. Compared with the transformations of Wu and Lafortune [17] and Balun and Masopust [15], which use neutral states and are polynomial in the system size and the value of K, our new transformations do not use neutral states and are polynomial in the size of the system and the encoding (logarithm) of K.

The need for the new transformations comes from the facts that (i) a large value of K makes the existing transformations infeasible, and (ii) the meaning of neutral states is unclear or questionable. Although we allow neutral states to appear in the systems, we neither use them nor create them in the transformations; using neutral states would result in transformations that do not work when the neutral states are not allowed [15].

Our motivation for the transformations is two-fold. First, they provide a deeper understanding of differences/similarities of the notions: we see that one secret state is sufficient for CSO, and we learn how to transform K-SO to K'-SO for any K and K'. Second, they are a tool to transfer complexity results among the notions: we get that deciding CSO for systems with a single secret state is as hard as deciding CSO for general systems, or that the existing complexity results for K-SO (and hence also INSO and CSO) [15, Table 1] hold for systems without neutral states and K given as part of the input.

To simplify the presentation of this section, some auxiliary technical results are moved to the appendices.

### A. Transforming CSO to K-SO for any $K \in \mathbb{N}_\infty$

The problem of deciding current-state opacity consists of a DES $G = (Q, \Sigma, \delta, I)$, secret states $Q_S$, non-secret states $Q_{NS}$, and projection $P: \Sigma^* \to \Sigma_o^*$. From $G$, we construct a DES $G' = (Q \cup \{q_s, q_{ns}\}, \Sigma \cup \{@\}, \delta', I)$ over the alphabet $\Sigma \cup \{@\}$, where $@$ is a new observable event, by adding two new states $q_s$ and $q_{ns}$. The transition function $\delta'$ of $G'$ is initialized as the transitions function $\delta$ of $G$ and further extended as follows, see Figure 4 for an illustration:

1) for every state $q \in Q_S$, we add $(q, @, q_s)$ to $\delta'$;
2) for every state $q \in Q_{NS}$, we add $(q, @, q_{ns})$ to $\delta'$.

We define $P': (\Sigma \cup \{@\})^* \to (\Sigma_o \cup \{@\})^*$, secret states $Q'_S = \{q_s\}$, and non-secret states $Q'_{NS} = Q_S \cup Q_{NS} \cup \{q_{ns}\}$. We now prove the correctness of the transformation.
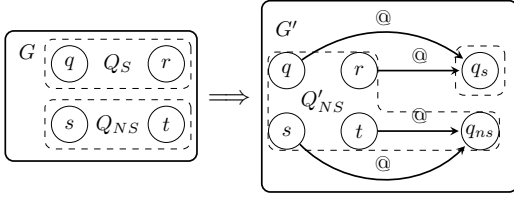
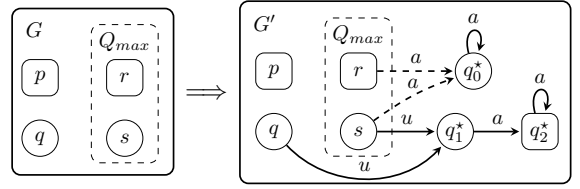Figure 4. Transforming CSO to K-SO for any K $\in \mathbb{N}_\infty$.



Figure 5. Transforming CSO to K-SO with a single observable event. Secret states are squared; the dashed transitions are in $G'$ only if $L(G)$ is finite.

**Theorem 4.** *The DES $G$ is CSO w.r.t. $Q_S$, $Q_{NS}$, $P$ iff the DES $G'$ is K-SO w.r.t. $Q'_S$, $Q'_{NS}$, $P'$.*

*Proof.* Assume that $G$ is not CSO. Then there is a string $w \in \Sigma^*$ that leads $G$ to a secret state, and every string that looks the same as $w$ leads $G$ out of non-secret states. Then, in $G'$, generating the string $w@$ ends up in the secret state $q_s \in \delta'(I, w@) \cap Q'_S \neq \emptyset$. Since generating any string that looks the same as $w$ leads $G$ to a state out of non-secret states, we have that $\delta'(I, P'^{-1}P'(w@)) \cap Q'_{NS} = \emptyset$. Therefore, $G'$ is not CSO, neither K-SO for any K $\in \mathbb{N}_\infty$.

Now, assume that $G$ is CSO, and let $st \in L(G')$ be such that $s$ leads $G'$ to a secret state and $t$ may be generated from this secret state in $G'$, formally $\delta'(\delta'(I, s) \cap Q'_S, t) \neq \emptyset$. Then, $s = s_1@$ where $s_1$ does not contain $@$, and $t = \varepsilon$. By construction, generating $s_1$ in $G$ ends up in a secret state. Since $G$ is CSO, there is a string $s'_1 \in P^{-1}P(s_1)$ looking the same as $s_1$ such that generating $s'_1$ in $G$ ends up in a non-secret state. Then, generating $s'_1@$ in $G'$ ends up in a non-secret state, and hence taking $s' = s'_1@$ gives that $P'(s't) = P'(s') = P'(s) = P'(st)$ and $\delta'(\delta'(I, s') \cap Q'_{NS}, t) \neq \emptyset$, which shows that $G'$ is K-SO for any K $\in \mathbb{N}_\infty$. $\square$

The transformation can be done in polynomial time, does not depend on K, and does not use neutral states. However, it introduces a new observable event.

To decrease the number of observable events in $G'$, we may notice that $G'$ is K-SO, for any K $\in \mathbb{N}_\infty$, if and only if $G'$ is CSO, since there are no transitions from the secret state $q_s$ of $G'$. Taking an encoding $e' \colon \Sigma_o \to \{0,1\}^k$ for a suitable $k$ (see Appendix A), and defining $e(a) = 0e'(a)$, for $a \in \Sigma_o$, and $e(@) = 1^{k+1}$, we get an encoding $e \colon \Sigma_o \cup \{@\} \to \{0,1\}^{k+1}$ that encodes observable events of $\Sigma$ as binary sequences starting with 0, and $@$ as a sequence of 1's. Applying the construction of Appendix A to $G'$ and $e$ results in $G''$ with two observable events, 0 and 1, the only secret state $q_s$, and no transitions from the secret state $q_s$, $G''$ is K-SO if and only if $G''$ is CSO, which is if and only if $G'$ is CSO by Lemma 12.

Notice that the transformation reduces CSO to K-SO with a single secret state, and hence we have the following corollary.

**Corollary 5.** *For any K $\in \mathbb{N}_\infty$, deciding K-step opacity of a system with a single secret state and two or more observable events is* PSPACE-*complete.* $\square$

If $G$ has a single observable event, the previous construction results in $G'$ with two observable events, and the construction of $G''$ does not work because the technique of Appendix A requires at least three observable events in $G'$. Therefore, we design a direct transformation preserving a single observable event that does not admit neutral states. For systems admitting neutral states, we refer to our recent work [15].

The problem of deciding CSO for systems with a single observable event consists of a DES $G = (Q, \Sigma, \delta, I)$ with $\Sigma_o = \{a\}$, secret states $Q_S$, non-secret states $Q_{NS} = Q \setminus Q_S$, and projection $P \colon \Sigma^* \to \{a\}^*$. From $G$, we construct a DES $G' = (Q \cup \{q_0^\star, q_1^\star, q_2^\star\}, \Sigma \cup \{u\}, \delta', I)$ by adding a new unobservable event $u$ and three new states $q_0^\star, q_1^\star, q_2^\star$. The transition function $\delta'$ is initialized as $\delta$ and further extend by adding, for each $q \in Q_{NS}$, the transition $(q, u, q_1^\star)$, and by adding three transitions $(q_1^\star, a, q_2^\star)$, $(q_0^\star, a, q_0^\star)$, and $(q_2^\star, a, q_2^\star)$; see Figure 5 for an illustration. Now, we determine (in linear time) whether the language $L(G)$ is finite. If so, we denote by $m = \max\{|P(w)| \mid w \in L(G)\}$ the maximal number of observable events in the strings of $L(G)$, and by $Q_{max} = \{q \in Q \mid q \in \delta(I, P^{-1}(a^m))\}$ the states reachable by the strings with the maximal number of observable events. Finally, we add the transition $(q, a, q_0^\star)$, for every $q \in Q_{max}$, to $\delta'$ and define $P' \colon (\Sigma \cup \{u\})^* \to \Sigma_o^*$, secret states $Q'_S = Q_S \cup \{q_2^\star\}$, and non-secret states $Q'_{NS} = Q_{NS} \cup \{q_0^\star, q_1^\star\}$, see Figure 5.

We first formulate a simple, but important, observation.

**Lemma 6.** *If $G$ is CSO, then, for every $w \in L(G)$, there exists $w' \in P^{-1}P(w)$ such that $\delta(I, w') \cap Q_{NS} \neq \emptyset$.*

*Proof.* For $w \in L(G)$, either $\delta(I, w) \cap Q_{NS} \neq \emptyset$ or $\delta(I, w) \subseteq Q_S$. In the latter case, CSO of $G$ implies that there is $w' \in P^{-1}P(w)$ such that $\delta(I, w') \cap Q_{NS} \neq \emptyset$. $\square$

We now prove the correctness of the construction.

**Theorem 7.** *The DES $G$ with a single observable event is CSO w.r.t. $Q_S$, $Q_{NS}$, $P$ iff $G'$ is K-SO w.r.t. $Q'_S$, $Q'_{NS}$, $P'$.*

*Proof.* Assume that $G$ is CSO. We show that if $st \in L(G')$ with $|P'(t)| \leq$ K and $s$ leads $G'$ to a secret state from which $t$ can be generated, then there are strings $s'$ and $t'$ with $P'(s') = P'(s)$ and $P'(t') = P'(t)$ such that $s'$ leads $G'$ to a non-secret state from which $t'$ can be generated.

If $s$ leads $G'$ to a secret state $q_s$, which is also a state of $G$, then CSO of $G$ implies that there is $s'$ with $P(s') = P(s)$ leading $G$, and hence $G'$, to a non-secret state, $q_{ns}$. Therefore, for any extension $t$ of $s$ from state $q_s$, $t' = ua^{|P'(t)|}$ is an extension of $s'$ from $q_{ns}$ with $P'(t') = P'(t)$.

If $s$ leads $G'$ to state $q_2^\star$, we have two cases. If $L(G)$ is infinite, there is $s'' \in L(G)$ with $P(s'') = P'(s)$. By Lemma 6, there is $s' \in L(G)$ with $P(s') = P(s'')$ that leads $G$, and hence $G'$, to a non-secret state, say $q_{ns}$. Then, $t' = ua^{|P'(t)|}$ is an extension of $s'$ from $q_{ns}$ with $P'(t') = P'(t)$. If $L(G)$ is finite, then $s = s_1 u s_2$, and $s'' = s_1 s_2$ satisfies $P(s'') = P'(s)$. If $s''$ leads $G'$ to state $q_0^\star$, any extension $t$ of $s$ from state $q_2^\star$
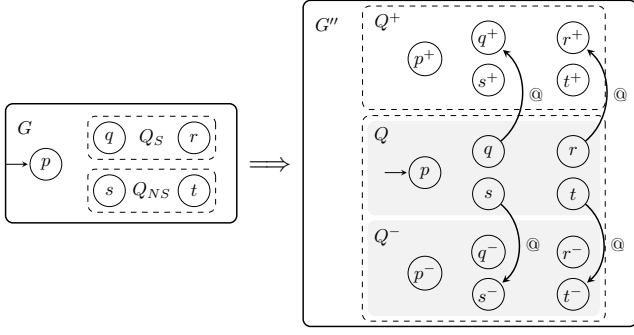
Figure 6. Automaton $G''$ of the first step transforming K-SO to CSO.

is an extension of $s' = s''$ from state $q_0^\star$. If $s'' \in L(G)$, then, by Lemma 6, there is $s'$ with $P(s') = P(s'')$ leading $G$ to a non-secret state, from which $t' = ua^{|t|}$ can be generated. Altogether, $G'$ is K-SO.

On the other hand, if $G$ is not CSO, there is $w \in L(G)$ such that $\delta(I, P^{-1}P(w)) \cap Q_S \neq \emptyset$ and $\delta(I, P^{-1}P(w)) \cap Q_{NS} = \emptyset$. In particular, $q_1^\star \notin \delta'(I, P'^{-1}P'(w)) \cap Q'_{NS}$. Since $w \in L(G)$, $|P(w)| \leq m$, the maximal number of $a$'s in the strings of $L(G)$, and hence $q_0^\star \notin \delta'(I, P'^{-1}P'(w)) \cap Q'_{NS}$. Altogether, $\delta'(I, P'^{-1}P'(w)) \cap Q'_{NS} = \emptyset$, and therefore $G'$ is not CSO, neither K-SO. $\qquad\square$

### B. Transforming K-SO to CSO

The problem of deciding K-step opacity consists of a DES $G = (Q, \Sigma, \delta, I)$, secret states $Q_S$, non-secret states $Q_{NS}$, and projection $P\colon \Sigma^* \to \Sigma_o^*$. From $G$, we first construct a DES $G'' = (Q \cup Q^+ \cup Q^-, \Sigma \cup \{@\}, \delta', I)$ by creating two disjoint copies of $G$, denoted by $G^+$ and $G^-$, with the state sets $Q^+ = \{q^+ \mid q \in Q\}$ and $Q^- = \{q^- \mid q \in Q\}$, and with an additional observable event @ that connects $G$ to $G^+$ and $G^-$ by the transitions $(q, @, q^+)$, for every $q \in Q_S$, and $(q, @, q^-)$, for every $q \in Q_{NS}$. The secret states are $Q''_S = Q^+$ and the non-secret states are $Q''_{NS} = Q \cup Q^-$, see Figure 6.

The idea of the construction is that if $G$ is K-SO, and hence CSO, then $G$ is in a non-secret state whenever it is in a secret state. Therefore, being in a secret (and hence also in a non-secret) state, the new @-transitions move the computation to both new copies $G^+$ and $G^-$. In these copies, we verify that if $G$ can make $k$ steps from the secret state (in $G^+$), it can also make $k$ steps from the corresponding non-secret state (in $G^-$). This is verified using current-state opacity, by considering the states of $G^+$ secret and of $G^-$ non-secret, which requires that every move in $G^+$ must be accompanied by a move in $G^-$.

Notice that $G''$ can be constructed in polynomial-time using no neutral states. The construction of $G''$ is already suitable to verify INSO of $G$ by checking CSO of $G''$.

**Theorem 8** (Transforming INSO to CSO). *The DES $G$ is INSO w.r.t. $Q_S$, $Q_{NS}$, and $P$ iff $G''$ is CSO w.r.t. $Q''_S$, $Q''_{NS}$, and $P''\colon (\Sigma \cup \{@\})^* \to (\Sigma_o \cup \{@\})^*$.*

*Proof.* Assume that $G$ is INSO. To show that $G''$ is CSO, we consider a string $w$ such that $\delta''(I, w) \cap Q''_S \neq \emptyset$, and show that there is $w'$ such that $P''(w) = P''(w')$ and $\delta''(I, w') \cap Q''_{NS} \neq \emptyset$. Since $Q''_S = Q^+$, string $w$ is of the form $w_1 @ w_2$.

By construction, there is a secret state $q \in \delta(I, w_1) \cap Q_S$ in $G$ such that $q^+ \in \delta''(I, w_1@) \cap Q''_S$ in $G''$, and $w_2$ is generated from $q^+$. Therefore, we can generate $w_2$ from state $q$ in $G$, that is, $\delta(\delta(I, w_1) \cap Q_S, w_2) \neq \emptyset$, and infinite-step opacity of $G$ implies that there is $w'_1 w'_2 \in L(G)$ such that $P(w_1) = P(w'_1)$, $P(w_2) = P(w'_2)$, and $\delta(\delta(I, w'_1) \cap Q_{NS}, w'_2) \neq \emptyset$. If we set $w' = w'_1 @ w'_2$, then $P''(w) = P''(w')$ and we have that $\emptyset \neq \delta''(\delta''(I, w'_1@) \cap Q''_{NS}, w'_2) \subseteq Q''_{NS}$, which completes this part of the proof.

If $G$ is not INSO, then there is a string $st \in L(G)$ such that $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$ and $\delta(\delta(I, s') \cap Q_{NS}, t') = \emptyset$ for every $s't' \in L(G)$ with $P(s) = P(s')$ and $P(t) = P(t')$. Taking $s@t \in L(G'')$, we obtain that $\emptyset \neq \delta''(\delta''(I, s@) \cap Q''_S, t) = \delta''(I, s@t) \subseteq Q''_S$ and, for every $s'@t' \in L(G'')$ with $P''(s@t) = P''(s'@t')$, we have that $\delta''(I, s'@t') \cap Q''_{NS} = \delta''(\delta''(I, s'@) \cap Q''_{NS}, t') = \emptyset$, and hence $G''$ is not CSO. $\qquad\square$

Although $G''$ can verify INSO of $G$ by checking CSO of $G''$, $G''$ is not suitable to verify K-SO in general; indeed, $G''$ verifies any number of steps from the visited secret state rather than at most K steps. To overcome this issue, we extend the construction by adding a counter that allows us to count up to K observable events from a visited secret state. To this aim, we use the automaton $\mathcal{A}_K$ constructed in Appendix B. Recall that $\mathcal{A}_K$ is of size polynomial in the logarithm of K, that the unique initial state of $\mathcal{A}_K$ is denoted by $q_0$, and that the observer of $\mathcal{A}_K$ has a unique path of length K consisting solely of non-marked states, while all the other states are marked.

However, the automata $G$, $G^+$, $G^-$ are over the alphabet $\Sigma$, while $\mathcal{A}_K$ is over $\Gamma$, which is disjoint from $\Sigma$. Therefore, we change the alphabets of the automata to $\Sigma' = \Sigma \cup (\Sigma_o \times \Gamma)$ as follows. In $G^+$ and $G^-$, we replace every *observable* transition $(p, \sigma, q)$ by $|\Gamma|$ transitions $(p, (\sigma, \gamma), q)$, for every $\gamma \in \Gamma$, and denote the results by $\tilde{G}^+$ and $\tilde{G}^-$. Similarly, in $\mathcal{A}_K$, we replace every transition $(p, \gamma, q)$ by $|\Sigma_o|$ transitions $(p, (\sigma, \gamma), q)$, for every observable $\sigma \in \Sigma_o$, and denote the result by $\tilde{\mathcal{A}}_K$. If we simplify the strings of the form $(a, a')(b, b')$ as $(ab, a'b')$, then the language $P(L(\tilde{G}^+)) = \{(w, w') \in (\Sigma_o \times \Gamma)^* \mid w \in P(L(G^+)), |w| = |w'|\}$. Similarly for $\tilde{G}^-$ and $\tilde{\mathcal{A}}_K$.

For a moment, we admit neutral states, and construct the NFA $G'''$ as a disjoint union of $G$, $\tilde{G}^+$, $\tilde{G}^-$, and $\tilde{\mathcal{A}}_K$, together with the transitions $(q, @, q^+)$ and $(q, @, q_0)$, for every $q \in Q_S$, where $q_0$ is the initial state of $\tilde{\mathcal{A}}_K$, and by $(q, @, q^-)$, for every $q \in Q_{NS}$. The secret states are $Q'''_S = Q^+$ and the non-secret states are $Q'''_{NS} = Q^- \cup \{\text{marked states of } \tilde{\mathcal{A}}_K\}$. The other states are *neutral*.

The construction transforms the K-SO problem of $G$ to the CSO problem of $G'''$, as we show below. Since the transformation is polynomial in both the system size and the encoding (logarithm) of K, it improves our recent result [15].

To illustrate the construction, we transform the 6-SO problem of $G = (\{1, \ldots, 8\}, \{a\}, \delta, \{1, 2\})$ with the transitions $\delta(i, a) = \{i + 1\}$, $i = 1, \ldots, 7$, $Q_S = \{1\}$, and $Q_{NS} = \{2\}$. Notice that $G$ is 6-SO, since we can make 6 steps from both states 1 and 2. To encode K = 6, the transformation uses NFA $\mathcal{A}_6 = \mathcal{A}_{2,2}$ (see Appendix B), and results in $G'''$ depicted in Figure 7, where all non-secret states are marked. The minimized observer of $G'''$ is shown in Figure 8. Since
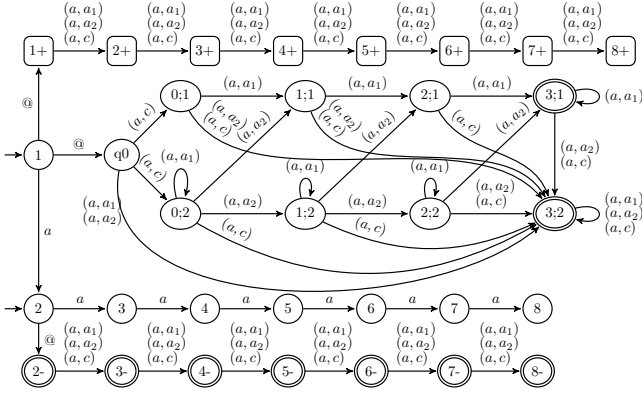
Figure 7. DES $G'''$ of the transformation 6-SO to CSO with neutral states; secret states are squared and non-secret states are marked.
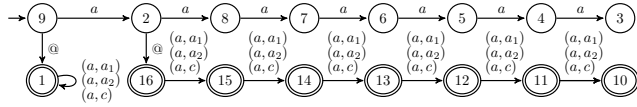


Figure 8. The minimized observer of $G'''$ of Figure 7.

every state of the observer reachable by a string containing @ is marked, it has to contain a non-secret state of $G$, that is, $G'''$ is CSO.

If we remove state $8$ from $G$ together with the corresponding transitions, then $G$ is not 6-SO, since we can make six steps from the secret state $1$, but only five steps from the corresponding non-secret state $2$. The transformation results in $G'''$ that coincides with the automaton of Figure 7 disregarding states $8, 8^+, 8^-$, and corresponding transitions. The minimized observer is shown in Figure 9, where state $4$, corresponding to state $\{7^+, (2;1), (2;2)\}$ consisting of secret states of $G$, is reachable by @$(a,c)(a,a_1)(a,a_1)(a,a_2)(a,a_1)(a,a_2)$, that is, $G'''$ is not CSO.

**Theorem 9** (K-SO to CSO with neutral states). *The DES $G$ is K-SO w.r.t. $Q_S$, $Q_{NS}$, and $P$ iff $G'''$ is CSO w.r.t. $Q'''_S$, $Q'''_{NS}$, and $P''' : (\Sigma' \cup \{@\})^* \to (\Sigma_o \cup \{@\} \cup \Sigma_o \times \Gamma)^*$.*

*Proof.* Assume that $G$ is K-SO. We show that $G'''$ is CSO. To this end, we consider a string $w$ such that $\delta'''(I, w) \cap Q'''_S \neq \emptyset$, and show that there is a string $w' \in P'''^{-1}P'''(w)$ such that $\delta'''(I, w') \cap Q'''_{NS} \neq \emptyset$. Since $Q'''_S = Q^+$, string $w$ is of the
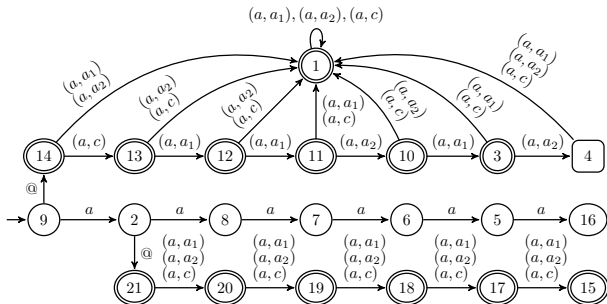


Figure 9. The minimized observer of $G'''$ of Figure 7 disregarding states $8$, $8^+$, $8^-$, and corresponding transitions.

form $w_1@w_2$ and, by construction, $\delta(I, w_1)$ contains a secret state of $G$ from which $w_2$ can be generated. If $|P(w_2)| \leq$ K, then K-SO of $G$ implies the existence of $w'_1 w'_2 \in L(G)$ such that $P(w'_1) = P(w_1)$, $P(w'_2) = P(w_2)$, and $\delta(\delta(I, w'_1) \cap Q_{NS}, w'_2) \neq \emptyset$; that is, there is a non-secret state $q \in \delta(I, w'_1)$ from which $w'_2$ can be generated, reaching a state $r$. Then, for $w' = w'_1@(w'_2, x)$, where $x$ is a prefix of the unique string not accepted by $\mathcal{A}_K$ of length $|P(w'_2)|$, we obtain that $\delta'''(I, w') \cap Q'''_{NS} \neq \emptyset$, since the non-secret state $r^- \in Q^-$ is reachable from state $q^-$ in $G'''$ by $(w'_2, x)$. If $|P(w_2)| >$ K, every string $w'_1@(w'_2, y) \in (\Sigma' \cup \{@\})^*$ is such that $y$ is accepted by $\mathcal{A}_K$, and hence $\delta'''(I, w'_1@(w'_2, y)) \cap Q'''_{NS} \neq \emptyset$. Thus, $G'''$ is CSO.

Assume that $G$ is not K-SO, that is, there exists $st \in L(G)$ 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 $t' \in P^{-1}P(t)$, $\delta(\delta(I, s') \cap Q_{NS}, t') = \emptyset$. Then, in particular, $\delta'''(I, s@) \cap Q'''_S \neq \emptyset$. If $\delta(I, s') \cap Q_{NS} = \emptyset$, then $\delta'''(I, s'@) \cap Q'''_{NS} = \emptyset$, and hence $G'''$ is not CSO. If $\delta(I, s') \cap Q_{NS} = Z \neq \emptyset$, we consider any string $s'@(t', y) \in L(G''')$, where $y$ is a prefix of the unique string not accepted by $\mathcal{A}_K$, which exists because $|y| = |P(t')| \leq$ K. Then, $(t', y)$ is not accepted by $\tilde{\mathcal{A}}_K$, and hence $\delta'''(I, s'@(t', y)) \cap Q'''_{NS} = \delta'''([\delta'''(I, s'@) \cap Q^-], (t', y)) = \delta'''(Z^-, (t', y)) = \emptyset$, where $Z^- = \{z^- \mid z \in Z\}$, because $(t', y)$ is not generated in $G'''$ from a state of $Z^-$, since $t'$ cannot be generated in $G$ from any $z \in Z$. Again, $G'''$ is not CSO. $\qquad\square$

Finally, to transform K-SO to CSO without using neutral states, we make all states of $\tilde{G}^+$ both initial and marked, and synchronize the computations of $\tilde{G}^+$ and $\tilde{\mathcal{A}}_K$ by their synchronous product $\tilde{G}^+ \| \tilde{\mathcal{A}}_K$. Now, we construct a DES $G'$ as a disjoint union of $G$, $\tilde{G}^-$, and $\tilde{G}^+ \| \tilde{\mathcal{A}}_K$, connected together by transitions $(q, @, (q^+, q_0))$, for every $q \in Q_S$, and $(q, @, q^-)$, for every $q \in Q_{NS}$. The secret states of $G'$ are the non-marked states of $\tilde{G}^+ \| \tilde{\mathcal{A}}_K$. All the other states are non-secret.

This transformation can be done in polynomial time in the system size and the binary encoding of K. How to reduce the number of observable events (in all transformations of this section) is discussed in Appendix A.

To illustrate the construction, we again transform the 6-SO problem $G = (\{1, \ldots, 8\}, \{a\}, \delta, \{1, 2\})$ with state $1$ secret and other states non-secret, and $\delta(i, a) = \{i+1\}$, $i = 1, \ldots, 7$. The transformation results in $G'$ depicted in Figure 10, using again the NFA $\mathcal{A}_6$. The minimized observer of $G'$ is depicted in Figure 11. Since every state of the observer reachable by a string containing @ is marked, it has to contain a non-secret state of $G$, that is, $G'$ is CSO.

If we remove state $8$ from $G$ together with the corresponding transitions, the transformation results in the DES $G'$ that coincides with the NFA of Figure 10 without states containing $8, 8^+, 8^-$, and the corresponding transitions. The minimized observer is shown in Figure 12, where state $4$, abbreviating the state $\{(7^+, (2;1)), (7^+, (2;2))\}$ consisting of secret states of $\tilde{G}^+ \| \mathcal{A}_K$, is reachable by the string @$(a,c)(a,a_1)(a,a_1)$ $(a,a_2)(a,a_1)(a,a_2)$, that is, $G'$ is not CSO.

**Theorem 10** (K-SO to CSO without neutral states). *The DES $G$ is K-SO w.r.t. $Q_S$, $Q_{NS}$, and $P$ iff $G'$ is CSO w.r.t. $Q'_S$, $Q'_{NS}$, and $P' : (\Sigma' \cup \{@\})^* \to (\Sigma_o \cup \{@\} \cup \Sigma_o \times \Gamma)^*$.*
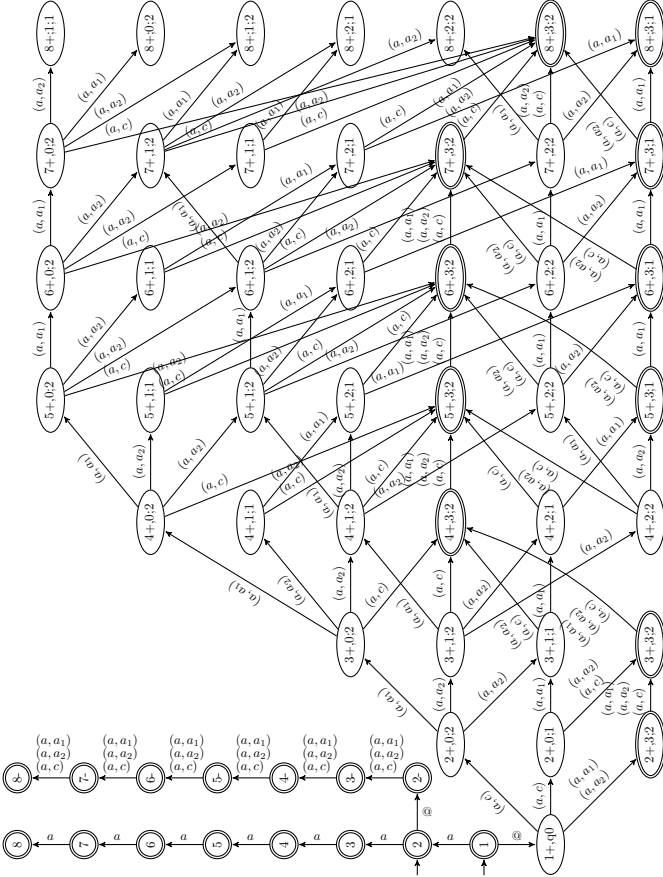
Figure 10. DES $G'$ with a relevant part of $\tilde{G}^+\|\tilde{\mathcal{A}}_6$; non-secret states are marked, other states are secret.
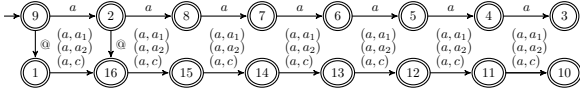


Figure 11. The minimized observer of $G'$.

*Proof.* Assume that $G$ is K-SO. We show that $G'$ is CSO. To this end, we consider a string $w$ such that $\delta'(I, w) \cap Q'_S \neq \emptyset$, and show that there exists a string $w' \in P'^{-1}P'(w)$ such that $\delta'(I, w') \cap Q'_{NS} \neq \emptyset$. Since $Q'_S$ consists of non-marked states of $\tilde{G}^+\|\tilde{\mathcal{A}}_K$, string $w$ is of the form $w_1@w_2$ and, by construction, $\delta(I, w_1)$ contains a secret state of $G$, from which $w_2$ can be generated. If $|P(w_2)| \leq K$, then K-SO of $G$ implies the existence of $w'_1w'_2 \in L(G)$ such that $P(w'_1) = P(w_1)$, $P(w'_2) = P(w_2)$, and $\delta(\delta(I, w'_1) \cap Q_{NS}, w'_2) \neq \emptyset$; that is, there is a non-secret state $q \in \delta(I, w'_1)$ from which $w'_2$ can be generated, reaching a state $r$. Then, for $w' = w'_1@(w'_2, x)$,
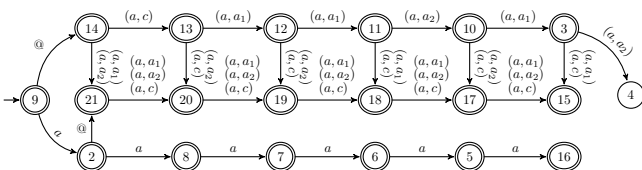


Figure 12. The minimized observer of $G'$ of Figure 10 disregarding states containing $8$, $8^+$, $8^-$, and corresponding transitions.

where $x$ is a prefix of the unique string not accepted by $\mathcal{A}_K$ of length $|P(w_2)|$, we obtain that $\delta'(I, w') \cap Q'_{NS} \neq \emptyset$, since the non-secret state $r^- \in Q^-$ is reachable from state $q^-$ in $G'$ by $(w'_2, x)$. If $|P(w_2)| > K$, every string $w'_1@(w'_2, y) \in (\Sigma' \cup \{@\})^*$ is such that $y$ is accepted by $\mathcal{A}_K$, and hence $(w'_2, y)$ is accepted by $\tilde{G}^+\|\tilde{\mathcal{A}}_K$ because $L(\tilde{G}^+\|\tilde{\mathcal{A}}_K) = L(\tilde{G}^+) \| L(\tilde{\mathcal{A}}_K)$ and $(w'_2, y)$ belongs to both $L(\tilde{G}^+)$ and $L(\tilde{\mathcal{A}}_K)$. Therefore, $\delta'(I, w'_1@(w'_2, y)) \cap Q'_{NS} \neq \emptyset$, and $G'$ is CSO.

Assume that $G$ is not K-SO, that is, there exists $st \in L(G)$ 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 $t' \in P^{-1}P(t)$, $\delta(\delta(I, s') \cap Q_{NS}, t') = \emptyset$. This implies that $\delta'(I, s@) \cap Q'_S \neq \emptyset$. If $\delta(I, s') \cap Q_{NS} = \emptyset$, then $\delta'(I, s'@) \cap Q'_{NS} = \emptyset$, and hence $G'$ is not CSO. If $\delta(I, s') \cap Q_{NS} = Z \neq \emptyset$, we take any $s'@(t', y) \in L(G')$, where $y$ is a prefix of the unique string not accepted by $\mathcal{A}_K$, which exists because $|y| = |P(t')| \leq K$. Then, $(t', y)$ is not accepted by $\tilde{G}^+\|\tilde{\mathcal{A}}_K$, and hence $\delta'(I, s'@(t', y)) \cap Q'_{NS} = \delta'([\delta'(I, s'@) \cap Q^-], (t', y)) = \delta'(Z^-, (t', y)) = \emptyset$, where $Z^- = \{z^- \mid z \in Z\}$, because $(t', y)$ is not generated in $G'$ from a state of $Z^-$, since $t'$ cannot be generated in $G$ from any $z \in Z$. Again, $G'$ is not CSO. $\square$

Again, we provide a direct transformation for systems with one observable event, where we do not admit neutral states; see our recent work [15] for systems admitting neutral states.

The K-SO problem for systems with one observable event consists of a DES $G = (Q, \Sigma, \delta, I)$ with $\Sigma_o = \{a\}$, secret states $Q_S$, non-secret states $Q_{NS} = Q \setminus Q_S$, and projection $P \colon \Sigma^* \to \{a\}^*$. We denote the number of states of $G$ by $n$, and determine (in linear time) whether $P(L(G))$ is finite.

If $P(L(G))$ is finite, we verify K-SO of $G$ in linear time by checking the subsets of states $\delta(I, P^{-1}(a^k))$, for $k \leq n - 1$. If $G$ is K-SO, and hence CSO, we set $Q'_S = Q_S$ and $Q'_{NS} = Q_{NS}$. If $G$ is not K-SO, we set $Q'_{NS} = \emptyset$ and $Q'_S = Q$.

If $P(L(G))$ is infinite, we define $Q'_{NS} = \{q \in Q_{NS} \mid \varphi(q) = K\}$, where $\varphi \colon Q \to \{0, \ldots, K\}$ assigns to state $q$ the maximal $k \in \{0, \ldots, K\}$ of observable steps possible from $q$. Formally, $\varphi(q) = \max\{k \in \{0, \ldots, K\} \mid \delta(q, P^{-1}(a^k)) \neq \emptyset\}$. The secret states are $Q'_S = Q \setminus Q'_{NS}$.

**Theorem 11** (K-SO to CSO with a single observable event). *The DES $G$ with a single observable event is K-SO w.r.t. $Q_S$, $Q_{NS}$, and $P$ iff $G$ is CSO w.r.t. $Q'_S$, $Q'_{NS}$, and $P$.*

*Proof.* Assume that $G$ is K-SO. If $P(L(G))$ is finite, then $G$ is CSO. If $P(L(G))$ is infinite, then, for every $w \in L(G)$, there is a state $q \in \delta(I, P^{-1}P(w))$ such that $\varphi(q) = K$. Since $G$ is K-SO, for every secret state $q_s \in \delta(I, P^{-1}P(w))$, there is a non-secret state $q_{ns} \in \delta(I, P^{-1}P(w))$ such that $\varphi(q_{ns}) \geq \varphi(q_s)$. Therefore, there is a non-secret state $q'_{ns} \in \delta(I, P^{-1}P(w))$ such that $\varphi(q'_{ns}) = K$, which means that $q'_{ns} \in Q'_{NS}$, and hence $G$ is CSO w.r.t. $Q'_{NS}$, $Q'_S$, and $P$.

Assume that $G$ is not K-SO. If $P(L(G))$ is finite, then $G$ is not CSO. If $P(L(G))$ is infinite, there is $w \in L(G)$ and a secret state $q_s \in \delta(I, P^{-1}P(w))$ such that $\varphi(q_s) > \varphi(q_{ns})$ for every non-secret state $q_{ns} \in \delta(I, P^{-1}P(w))$. Therefore, $\varphi(q_{ns}) < K$ for every $q_{ns} \in \delta(I, P^{-1}P(w)) \cap Q_{NS}$, and hence $\delta(I, P^{-1}P(w)) \cap Q'_{NS} = \emptyset$, which shows that $G$ is not CSO w.r.t. $Q'_S$, $Q'_{NS}$, and $P$. $\square$
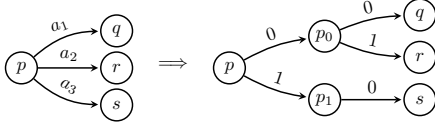
Figure 13. Replacement of observable events $a_1, a_2, a_3$ encoded $e(a_1) = 00$, $e(a_2) = 01$, and $e(a_3) = 10$, and new states $p_0$ and $p_1$.

## V. CONCLUSIONS

We designed a new algorithm verifying K-step opacity with better complexity than that of existing algorithms. In addition, our complexity does not depend on K. We compared K-step opacity with current-state opacity and infinite-step opacity, and provided new transformations among these notions that do not use neutral states, that are polynomial w.r.t. both the size of the system and the binary encoding of K, preserve determinism (see Appendix C), and the resulting systems of which do not have more observable events than the input systems.

## APPENDIX A
### REDUCING THE NUMBER OF OBSERVABLE EVENTS

We now discuss how to reduce the number of observable events in systems with at least three observable events without affecting the property of being CSO. This construction is a modification of the construction of Balun and Masopust [15].

For an NFA $G = (Q, \Sigma, \delta, I, F)$, an alphabet $\Gamma_o \subseteq \Sigma_o$ with at least three events, and a binary encoding $e \colon \Gamma_o \to \{0,1\}^k$ of the events of $\Gamma_o$, where $k \leq \lceil \log_2(|\Gamma_o|) \rceil + 1$, we define the NFA $G' = (Q', (\Sigma - \Gamma_o) \cup \{0,1\}, \delta', I, F)$ as follows. We replace every transition $(p, a, q)$ with $a \in \Gamma_o$ and $e(a) = b_1 b_2 \cdots b_k \in \{0,1\}^k$ by $k$ transitions

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

where $p_{b_1}, \ldots, p_{b_1 \cdots b_{k-1}}$ are states added to the state set of $G'$ as non-secret states. These states are created when needed for the first time, and reused later during the replacements. Figure 13 illustrates the replacement of three observable events $a_1, a_2, a_3$ with the encoding $e(a_1) = 00$, $e(a_2) = 01$, and $e(a_3) = 10$. Notice that $G'$ can be constructed from $G$ in polynomial time.

**Lemma 12.** *System $G$ is CSO w.r.t. $Q_S$, $Q_{NS}$, and $P \colon \Sigma^* \to \Sigma_o^*$ iff $G'$ is CSO w.r.t. $Q_S$, $Q_{NS} \cup (Q' - Q)$, and $P' \colon [(\Sigma - \Gamma_o) \cup \{0,1\}]^* \to [(\Sigma_o - \Gamma_o) \cup \{0,1\}]^*$.*

*Proof.* To show that $G$ is CSO iff $G'$ is CSO, we need to show that $P(L_S) \subseteq P(L_{NS})$ iff $P'(L'_S) \subseteq P'(L'_{NS})$ [15], where

- $L_S = L_m(Q, \Sigma, \delta, I, Q_S)$,
- $L_{NS} = L_m(Q, \Sigma, \delta, I, Q_{NS})$,
- $L'_S = L_m(Q', (\Sigma - \Gamma_o) \cup \{0,1\}, \delta', I, Q_S)$, and
- $L'_{NS} = L_m(Q', (\Sigma - \Gamma_o) \cup \{0,1\}, \delta', I, Q'_{NS})$, for $Q'_{NS} = Q_{NS} \cup (Q' - Q)$.

We define a morphism $f \colon \Sigma^* \to ((\Sigma - \Gamma_o) \cup \{0,1\})^*$ such 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 $G'$, $w \in L(G)$ iff $f(w) \in L(G')$. In particular, $P(w) \in P(L_S)$ iff $P'(f(w)) \in P'(L'_S)$, and $P(w) \in P(L_{NS})$ iff $P'(f(w)) \in P'(L'_{NS})$. Therefore, if $P'(L'_S) \subseteq P'(L'_{NS})$ then $P(L_S) \subseteq$

TABLE I
STRINGS $W_{k,n}$ USED IN THE PROOF OF LEMMA 13.

| $k \backslash n$ | 1 | 2 | 3 |
|---|---|---|---|
| 1 | $a_1$ | $a_1 a_2$ | $a_1 a_2 a_3$ |
| 2 | $a_1^2$ | $a_1^2 a_2 a_1 a_2$ | $a_1^2 a_2 a_1 a_2 a_3 a_1 a_2 a_3$ |
| 3 | $a_1^3$ | $a_1^3 a_2 a_1^2 a_2 a_1 a_2$ | $a_1^3 a_2 a_1^2 a_2 a_1 a_2 a_3 a_1^2 a_2 a_1 a_2 a_3 a_1 a_2 a_3$ |

$P(L_{NS})$. On the other hand, assume that $P(L_S) \subseteq P(L_{NS})$, and consider any $P'(x) \in P'(L'_S)$. Then, $P'(x)$ is of the form $P'(f(y))$ for some $y \in L_S$, and $P(y) \in P(L_S) \subseteq P(L_{NS})$ implies that $P'(x) = P'(f(y)) \in P'(L'_{NS})$. □

## APPENDIX B
### LOGARITHMIC ENCODING OF A K-STEP COUNTER

In this appendix, we construct an NFA $\mathcal{A}_K$ of size polynomial in the logarithm of K such that the observer of $\mathcal{A}_K$ has a unique path of length K consisting solely of non-marked states, while all the other states are marked. This path plays the role of a K-step counter that is essential in the transformation from K-SO to CSO of Section IV-B. To construct the automaton $\mathcal{A}_K$, we make use of NFAs $\mathcal{A}_{k,n}$, for every $k, n \geq 1$, that can be constructed in time polynomial in $k$ and $n$ and that are similar to NFAs we used earlier [23], though we need to adjust them.

**Lemma 13.** *For every integers $k, n \geq 1$, there is an NFA $\mathcal{A}_{k,n}$ with $n$ events and $n(k+2)$ states, such that $\mathcal{A}_{k,n}$ accepts all strings except for all prefixes of a unique string $W_{k,n}$, which is a string of length $\binom{k+n}{k} - 1$.*

*Proof.* For $k, n \geq 1$, we define $W_{k,n}$ over $\Sigma_n = \{a_1, \ldots, a_n\}$ by setting $W_{k,1} = a_1^k$, $W_{1,n} = a_1 a_2 \cdots a_n$, and

$$
\begin{aligned}
W_{k,n} &= W_{k,n-1} \, a_n \, W_{k-1,n} \\
&= W_{k,n-1} \, a_n \, W_{k-1,n-1} \, a_n \, W_{k-2,n} \quad (1) \\
&= W_{k,n-1} \, a_n \, W_{k-1,n-1} \, a_n \, \cdots \, a_n \, W_{1,n-1} \, a_n \, .
\end{aligned}
$$

The construction is illustrated in Table I. The length of $W_{k,n}$ is $\binom{k+n}{k} - 1$, and $a_n$ appears exactly $k$ times in $W_{k,n}$ [24]. For defining $\mathcal{A}_{k,n}$, it is useful to set $W_{k,n} = \varepsilon$ whenever $kn = 0$.

We construct an NFA $\mathcal{A}_{k,n}$ over $\Sigma_n$ marking $\Sigma_n^* - \overline{\{W_{k,n}\}}$. For $k \geq 0$, $\mathcal{A}_{k,1}$ is the minimal DFA marking $\{a_1\}^* - \{a_1^k\}$, consisting of $k + 2$ states of the form $(i; 1)$, see Figure 14, together with the given transitions. State $(k+1; 1)$ is marked, state $(0; 1)$ is initial.
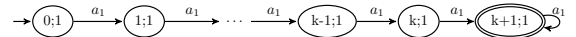


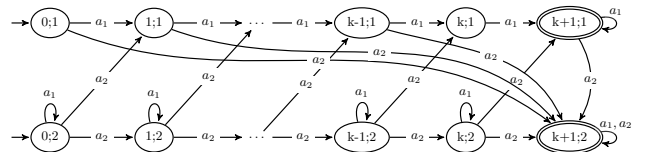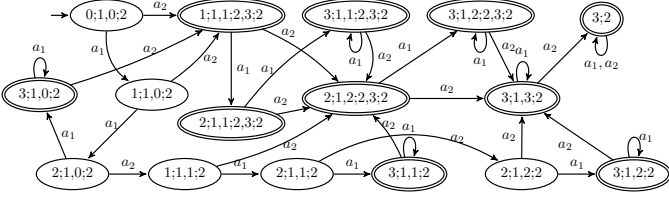Figure 14. The NFA $\mathcal{A}_{k,1}$ with $k+2$ states.



Figure 15. The NFA $\mathcal{A}_{k,2}$ with $2(k+2)$ states.

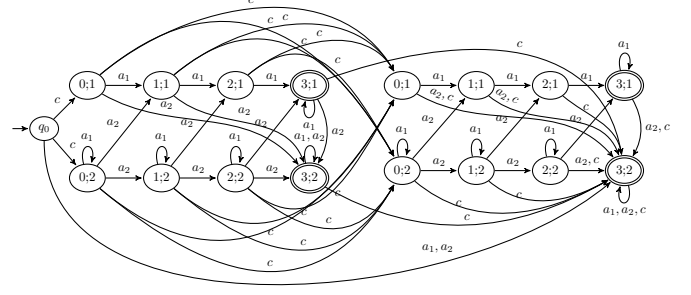Figure 16. The observer of the NFA $\mathcal{A}_{2,2}$.



Figure 17. Example for K = 12, which gives $a_2 = 2$, the automaton $\mathcal{A}_{12}$ consisting of two copies of $\mathcal{A}_{2,2}$.

Given $\mathcal{A}_{k,n-1}$, we construct $\mathcal{A}_{k,n}$ from $\mathcal{A}_{k,n-1}$ by adding $k+2$ states $(0;n),(1;n),\ldots,(k+1;n)$, where $(0;n)$ is added to initial, and $(k+1;n)$ to final states, see Figure 15 illustrating the construction for $n=2$; $\mathcal{A}_{k,n}$ has $n(k+2)$ states. We call the state $(k+1,n)$ *maximal*. Additional transitions of $\mathcal{A}_{k,n}$ consist of four groups:

1) Self-loops $(i;n) \xrightarrow{a_j} (i;n)$ for every $i \in \{0,\ldots,k+1\}$ and $a_j \in \{a_1,\ldots,a_{n-1}\}$;
2) Transitions $(i;n) \xrightarrow{a_n} (i+1;n)$ for every $i \in \{0,\ldots,k\}$, and the self-loop $(k+1;n) \xrightarrow{a_n} (k+1;n)$;
3) Transitions $(i;n) \xrightarrow{a_n} (i+1;m)$ for every $i \in \{0,\ldots,k\}$ and $m \in \{1,\ldots,n-1\}$;
4) Transitions $(i;m) \xrightarrow{a_n} (k+1;n)$ for every state $(i;m)$ of $\mathcal{A}_{k,n-1}$ with $i \neq k$.

The additional states of $\mathcal{A}_{k,n}$ and transitions (1) and (2) ensure marking of all strings containing more than $k$ events $a_n$. The transitions (3) and (4) ensure marking of all strings in $(\Sigma_{n-1}^* a_n)^{i+1} L(\mathcal{A}_{k-(i+1),n-1}) a_n \Sigma_n^*$ for which the string between the $(i+1)$-st and the $(i+2)$-nd occurrence of $a_n$ is not of the form $W_{k-(i+1),n-1}$, and hence not a correct substring of $W_{k,n} = W_{k,n-1} a_n \cdots a_n W_{k-(i+1),n-1} a_n \cdots a_n W_{1,n-1} a_n$. The transitions (4) ensure that all strings with a prefix $w a_n$, where $w$ is any string from $\Sigma_{n-1}^* - \{W_{k,n-1}\}$, are marked. Together, these conditions ensure that $\mathcal{A}_{k,n}$ marks every string that is not a prefix of $W_{k,n}$.

It remains to show that $\mathcal{A}_{k,n}$ does not mark any prefix of $W_{k,n}$, which we show by induction on $(k,n)$. For $(0,n)$, $n \geq 1$, string $W_{0,n} = \varepsilon$ is not marked by $\mathcal{A}_{0,n}$, since the initial states $(0,m) = (k,m)$ of $\mathcal{A}_{0,n}$ are not marked. Likewise, for $(k,1)$, $k \geq 0$, we find that the prefixes of $W_{k,1} = a_1^k$ are not marked by $\mathcal{A}_{k,1}$ (cf. Figure 14). For the inductive case $(k,n) \geq (1,2)$, where $\leq$ is the standard product order, we assume that $\mathcal{A}_{k',n'}$ marks no prefix of $W_{k',n'}$ for any $(k',n') < (k,n)$ and that $W_{k',n'}$ leads $\mathcal{A}_{k',n'}$ only to states of the form $(k;\cdot)$. Then, $W_{k,n} = W_{k,n-1} a_n W_{k-1,n}$, and no prefix of $W_{k,n-1}$ is marked by $\mathcal{A}_{k,n-1}$ by induction. In addition, there is no transition under $a_n$ from a state $(k;m)$ with $m \neq n$ in $\mathcal{A}_{k,n}$. Therefore, if a prefix of $W_{k,n}$ is marked by $\mathcal{A}_{k,n}$, it must be marked in a run starting from the initial state $(0;n)$. Since $W_{k,n-1}$ contains no $a_n$, we find that $\mathcal{A}_{k,n}$ can only reach states $\delta((0;n), W_{k,n-1} a_n) = \{(1;m) \mid 1 \leq m \leq n\}$ after generating $W_{k,n-1} a_n$, which are the initial states of $\mathcal{A}_{k-1,n}$. By induction, $\mathcal{A}_{k-1,n}$ marks no prefix of $W_{k-1,n}$, and hence no prefix of $W_{k,n}$ is marked by $\mathcal{A}_{k,n}$. $\qquad\square$

To illustrate the construction, we consider $k = n = 2$. Then, $W_{2,2} = a_1^2 a_2 a_1 a_2$, the NFA $\mathcal{A}_{2,2}$ has 8 states, and the observer of $\mathcal{A}_{2,2}$, shown in Figure 16, contains a unique path of length

$\binom{4}{2} - 1 = 5$ consisting solely of non-marked states while all the other states are marked.

We now show how to use the NFAs $\mathcal{A}_{i,j}$ to construct an automaton $\mathcal{A}_K$ of size polynomial in the logarithm of K such that the observer of $\mathcal{A}_K$ has a unique path consisting solely of non-marked states, while all the other states are marked. For simplicity, and without loss of generality, we only use the automata of the form $\mathcal{A}_{j,j}$.

Since $\binom{2n+2}{n+1} = \frac{4n+2}{n+1}\binom{2n}{n}$ and $\binom{2n}{n} \leq 4^n$, every natural number K can be expressed as

$$K = b_n \binom{2n}{n} + b_{n-1} \binom{2n-2}{n-1} + \cdots + b_1 \binom{2}{1} + b_0$$

for some $n \leq \lceil \log_4(K+1) \rceil$ and $b_i \in \{0,1,2,3\}$, $i = 0,\ldots,n$. This expression is not unique, e.g., K = 2 can be expressed as $b_0 = 2$, or $b_1 = 1$ and $b_0 = 0$.

For every $b_i$, $i = n,\ldots,0$, we create $b_i$ copies of $\mathcal{A}_{i,i}$ over $\Sigma_i = \{a_1 \ldots, a_i\}$, which results in a sequence of automata $\mathcal{B}_1,\ldots,\mathcal{B}_\ell$. We take a new event $c \notin \Sigma_n$ and connect all the automata $\mathcal{B}_1,\ldots,\mathcal{B}_\ell$ to a single automaton $\mathcal{A}_K$ by $c$-transitions as follows. For $j = 1,\ldots,\ell-1$, we add a $c$-transition from every non-marked state of $\mathcal{B}_j$ to every initial state of $\mathcal{B}_{j+1}$; from all the other states, the $c$-transition goes to the maximal state of $\mathcal{B}_\ell$. Finally, we add a new state, $q_0$, which is the only initial state of the automaton $\mathcal{A}_K$, $c$-transitions from $q_0$ to all initial states of $\mathcal{B}_1$, and transitions under all the other events to the maximal state of $\mathcal{B}_\ell$; see an illustrative example below.

Then, the observer of $\mathcal{A}_K$ has a unique path consisting of non-marked states along the string

$$(cW_{n,n})^{b_n} (cW_{n-1,n-1})^{b_{n-1}} \cdots (cW_{0,0})^{b_0}$$

of length $K = b_n \binom{2n}{n} + b_{n-1}\binom{2n-2}{n-1} + \cdots + b_0\binom{0}{0}$, and the other states are marked. Since every $\mathcal{B}_j$ is of size polynomial in $n$, we obtain that $\mathcal{A}_K$ is of size polynomial in the logarithm of K and its observer has a unique path of length K consisting solely of non-marked states, with all the other states marked.

**Lemma 14.** *For every natural number K, there is an automata $\mathcal{A}_K$ of size polynomial in $O(\log K)$ such that the observer of $\mathcal{A}_K$ has a unique path of length K consisting solely of non-marked states, and with all the other states marked.* $\qquad\square$

For an illustration, consider $K = 12 = 2\binom{4}{2} + 0\binom{2}{1} + 0$. We create two copies of $\mathcal{A}_{2,2}$ and connect them by $c$-transitions as shown in Figure 17. The observer with the unique path of non-marked states of length K = 12 is shown in Figure 18.
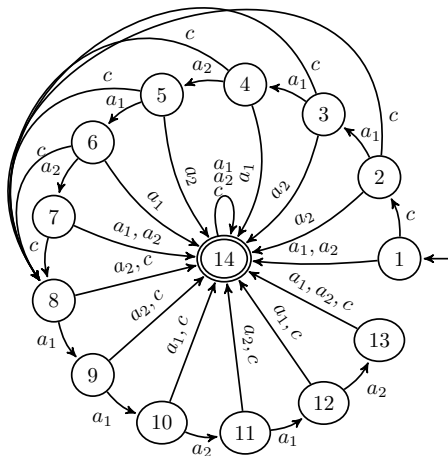
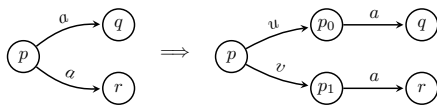Figure 18. The min. DFA of the observer with the unique path of length 12.



Figure 19. Determinization of an NFA.

## APPENDIX C
## PRESERVING DETERMINISM

In this section, we show how to make an NFA deterministic without affecting the property of being K-step opaque, for any $K \in \mathbb{N}_\infty$, by adding a few unobservable events.

From an NFA $G = (Q, \Sigma, \delta, I, F)$, we construct a DFA $G' = (Q', \Sigma', \delta', I, F)$ as follows. For every state $p$ and an event $a$ with $|\delta(p, a)| > 1$, we replace every transition $(p, a, q)$ of $G$ with two transitions $(p, u, p')$ and $(p', a, q)$, where $p'$ is a new state and $u$ is a new unobservable event (neither $p'$ nor $u$ are reused), see Figure 19 for an illustration. The secret status of the new state $p'$ is set according to the status of state $p$, that is, $p'$ is secret iff $p$ is. Notice that $G'$ can be constructed from $G$ in polynomial time.

**Lemma 15.** *System $G$ is K-SO w.r.t. $Q_S$, $Q_{NS}$, and $P \colon \Sigma^* \to \Sigma_o^*$ iff $G'$ is K-SO w.r.t. $Q'_S$, $Q'_{NS}$, and $P' \colon \Sigma'^* \to \Sigma_o^*$.*

*Proof.* Indeed, the number of observable steps from a state of $G$ is preserved in $G'$. Thus, we need to show that $G'$ is CSO iff $G$ is. However, every newly added state $p'$ is reachable by an unobservable event from its original state $p$, and hence $p'$ is contained in every state of the observer that contains $p$; and vice versa, because every path to state $p'$ goes through state $p$ in $G'$. Therefore, if a state of the observer contains a secret state $p'$ and a non-secret state $r'$, then it also contains the original secret state $p$ and the original non-secret state $r$. That is, $G$ is K-SO iff $G'$ is K-SO. □

## REFERENCES

[1] S. Schneider and A. Sidiropoulos, "CSP and anonymity," in *Computer Security — ESORICS 96*. Springer, 1996, pp. 198–218.

[2] N. BenHadj-Alouane, S. Lafrance, F. Lin, J. Mullins, and M. Yeddes, "On the verification of intransitive noninterference in mulitlevel security," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 5, pp. 948–958, 2005.

[3] R. Alur, P. Černý, and S. Zdancewic, "Preserving secrecy under refinement," in *Automata, Languages and Programming*. Springer, 2006, pp. 107–118.

[4] R. Focardi and R. Gorrieri, "A taxonomy of trace-based security properties for CCS," in *The Computer Security Foundations Workshop VII*. IEEE Comput. Soc. Press, 1994, pp. 126–136.

[5] L. Mazaré, "Decidability of opacity with non-atomic keys," in *Formal Aspects in Security and Trust*. Springer, 2004, pp. 71–84.

[6] J. W. Bryans, M. Koutny, and P. Y. Ryan, "Modelling opacity using Petri nets," *Electronic Notes in Theoretical Computer Science*, vol. 121, pp. 101–115, 2005.

[7] J. W. Bryans, M. Koutny, L. Mazaré, and P. Y. A. Ryan, "Opacity generalised to transition systems," *International Journal of Information Security*, vol. 7, no. 6, pp. 421–435, 2008.

[8] A. Saboori and C. N. Hadjicostis, "Notions of security and opacity in discrete event systems," in *IEEE Conference on Decision and Control*. IEEE, 2007, pp. 5056–5061.

[9] E. Badouel, M. Bednarczyk, A. Borzyszkowski, B. Caillaud, and P. Darondeau, "Concurrent secrets," *Discrete Event Dynamic Systems*, vol. 17, no. 4, pp. 425–446, 2007.

[10] J. Dubreil, P. Darondeau, and H. Marchand, "Opacity enforcing control synthesis," in *Workshop on Discrete Event Systems*. IEEE, 2008, pp. 28–35.

[11] R. Jacob, J.-J. Lesage, and J.-M. Faure, "Overview of discrete event systems opacity: Models, validation, and quantification," *Annual Reviews in Control*, vol. 41, pp. 135–146, 2016.

[12] A. Saboori and C. N. Hadjicostis, "Verification of infinite-step opacity and complexity considerations," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1265–1269, 2012.

[13] X. Yin and S. Lafortune, "A new approach for the verification of infinite-step and K-step opacity using two-way observers," *Automatica*, vol. 80, pp. 162–171, 2017.

[14] H. Lan, Y. Tong, J. Guo, and A. Giua, "Comments on "A new approach for the verification of infinite-step and K-step opacity using two-way observers" [Automatica 80 (2017) 162–171]," *Automatica*, vol. 122, p. 109290, 2020.

[15] J. Balun and T. Masopust, "Comparing the notions of opacity for discrete-event systems," *Discrete Event Dynamic Systems*, 2021.

[16] A. Wintenberg, M. Blischke, S. Lafortune, and N. Ozay, "A general language-based framework for specifying and verifying notions of opacity," *CoRR*, vol. 2103.10501, 2021.

[17] Y.-C. Wu and S. Lafortune, "Comparative analysis of related notions of opacity in centralized and coordinated architectures," *Discrete Event Dynamic Systems*, vol. 23, no. 3, pp. 307–339, 2013.

[18] C. G. Cassandras and S. Lafortune, Eds., *Introduction to Discrete Event Systems*, 2nd ed. Springer, 2008.

[19] J. E. Hopcroft, R. Motwani, and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 2006.

[20] G. Jirásková and T. Masopust, "On a structural property in the state complexity of projected regular languages," *Theoretical Computer Science*, vol. 449, pp. 93–105, 2012.

[21] K. Wong, "On the complexity of projections of discrete-event systems," in *Workshop on Discrete Event Systems*, 1998, pp. 201–206.

[22] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. MIT Press, 2009.

[23] M. Krötzsch, T. Masopust, and M. Thomazo, "Complexity of universality and related problems for partially ordered NFAs," *Information and Computation*, vol. 255, pp. 177–192, 2017.

[24] T. Masopust and M. Thomazo, "On boolean combinations forming piecewise testable languages," *Theoretical Computer Science*, vol. 682, pp. 165–179, 2017.