


# Speed Me up if You Can: Conditional Lower Bounds on Opacity Verification

Jiří Balun ✉ 

Faculty of Science, Palacky University Olomouc, Czechia

Tomáš Masopust ✉ 

Faculty of Science, Palacky University Olomouc, Czechia

Petr Osička ✉ 

Faculty of Science, Palacky University Olomouc, Czechia

---

## Abstract

Opacity is a property of privacy and security applications asking whether, given a system model, a passive intruder that makes online observations of system’s behaviour can ascertain some “secret” information of the system. Deciding opacity is a PSPACE-complete problem, and hence there are no polynomial-time algorithms to verify opacity under the assumption that PSPACE differs from PTIME. This assumption, however, gives rise to a question whether the existing exponential-time algorithms are the best possible or whether there are faster, sub-exponential-time algorithms. We show that under the (Strong) Exponential Time Hypothesis, there are no algorithms that would be significantly faster than the existing algorithms. As a by-product, we obtained a new conditional lower bound on the time complexity of deciding universality (and therefore also inclusion and equivalence) for nondeterministic finite automata.

**2012 ACM Subject Classification** Theory of computation → Formal languages and automata theory

**Keywords and phrases** Finite automata, opacity, fine-grained complexity

**Funding** *Jiří Balun*: Supported by IGA PrF 2023 026.

## 1 Introduction

In privacy and security applications and communication protocols, it is desirable to keep some information about the system or its behaviour secret. Such requirements put additional restrictions on the information flow of the system and have widely been discussed in the literature as various properties, including *anonymity* [14, 32, 38], *non-interference* [8, 12, 19, 33], *secrecy* [1, 3, 10], *security* [17], *perfect security* [46], and *opacity* [20, 30]. *Anonymity* is the property to preserve secrecy of identity of actions; for instance, web servers should not be able to learn the true source of a request. *Non-interference* asks whether, given two input states of the system that share the same values of specified variables, the behaviors of the system started from these states are indistinguishable by the observer under the observation of the specified variables. *Secrecy* expresses whether an observer can ever find out that a trajectory of the system belongs to a set of secret trajectories, and *perfect security* requires that an observer that knows the set of all trajectories of the system cannot deduce any information about occurrences of high-security events by observing low-security events.

In this paper, we are interested in (various types of) opacity, which in a sense generalizes the other mentioned properties; namely, the properties above can be verified by reduction to opacity. More specifically, Alur et al. [1] have shown that *secrecy* captures *non-interference* and *perfect security*, and Lin [26] has provided an extensive discussion and comparison of all these properties. He has in particular shown that *anonymity* and *secrecy*, and the properties of *observability* [27], *diagnosability* [25, 37], and *detectability* [28, 29, 39] of discrete-event systems are special cases of opacity. Wu et al. [44] and Góes et al. [18] discuss applications of opacity in location privacy, and Wintenberg et al. [42] apply opacity in contact tracing.

Given a system model, *opacity* asks whether a malicious passive observer (an intruder) with a complete knowledge of the structure of the model can ascertain some “secret” information of the system by making online incomplete observations of system’s behaviour. The secret information is modeled either as a set of states or as a set of behaviours of the system. Based on the incomplete observations, the intruder estimates the state/behaviour of the system, and the system is opaque if for every secret state/behaviour of the system, there is a non-secret state/behaviour of the system that looks the same to the intruder. If the secret information is given as a set of states, we talk about state-based opacity [9, 11], whereas if the secret information is given as a set of secret behaviours (a language), we talk about language-based opacity [4, 15]. Several notions of opacity have been discussed in the literature for systems modeled by automata and Petri nets; see Jacob et al. [24] for an overview. In this paper, we focus on finite automata models and on the notions of opacity that we review in Section 3.

The fastest existing algorithms verifying the notions of opacity under consideration have exponential-time complexity with respect to the number of states of the automaton. In fact, the verification of opacity is a PSPACE-complete problem [5, 6, 24], and hence we may conclude that there are no polynomial-time algorithms deciding opacity unless PTIME = PSPACE. Although the assumption that PTIME  $\neq$  PSPACE excludes the existence of polynomial-time algorithms, the question whether there is a significantly faster (i. e., sub-exponential-time) algorithm remains open.

To achieve stronger lower bounds (although still conditional), we use the *Exponential Time Hypothesis* (ETH) and its strong version—the *Strong Exponential Time Hypothesis* (SETH). Both hypotheses were formulated by Impagliazzo and Paturi [23] and are based on the observation that (so far) we were not able to find algorithms that would, in the worst case, solve SAT significantly faster than the algorithms trying all possible truth assignments. In particular, ETH states that 3-SAT cannot be solved in time  $2^{o(n)}$  where  $n$  is the number of variables. However, it admits algorithms solving 3-SAT in time  $O(c^n)$  where  $c < 2$ . In fact, the current fastest 3-SAT algorithm of Paturi et al. [31], improved by Hertli [21], runs in time  $O^*(1.30704^n)$ . With increasing  $k$ , the current fastest  $k$ -SAT algorithms are getting slower; for instance, the best 4-SAT algorithm of Hertli [21] runs in time  $O^*(1.46899^n)$ . This observation motivated the formulation of SETH that claims that, for any constant  $c < 2$ , there is always a sufficiently large  $k$  such that  $k$ -SAT cannot be solved in time  $O(c^n)$  [23]. Both hypotheses imply that the complexity classes PTIME and NP are separated; moreover, SETH implies ETH.

In this paper, we show that under (S)ETH, there are no significantly faster algorithms verifying opacity. In particular, we show that unless SETH fails, there is no algorithm that decides whether a given  $n$ -state automaton satisfies the considered notions of opacity and runs in time  $O^*(2^{n/c})$ , for any  $c > 2$  (Theorem 8 and Corollary 15). Since the number of symbols in the alphabet of our construction is unbounded and the standard binary encoding of symbols does not work under SETH, it is not clear whether this result also holds for automata with a fixed size (binary) alphabet. We partially explore this question under ETH rather than SETH. We show that unless ETH fails, there is no algorithm that decides whether a given  $n$ -state automaton (over a binary alphabet) satisfies the considered notions of opacity and runs in time  $O^*(2^{o(n)})$ . Our results are summarized in Table 1; for the complexity upper bounds, we refer the reader to the literature [6, 34].

As a by-product, we obtain a new conditional lower bound for deciding universality (and hence inclusion and equivalence) for nondeterministic automata (NFA): Unless SETH fails, there is no  $c > 2$  such that the universality of an  $n$ -state NFA can be decided in time  $O^*(2^{n/c})$

	Lower bound		Upper bound	
	$\Gamma$ not fixed	$ \Gamma  = 2$		
LBO	$O^*(2^{n/(2+\varepsilon)})$	$O^*(2^{o(n)})$	$O^*(2^n)$	
CSO	$O^*(2^{n/(2+\varepsilon)})$	$O^*(2^{o(n)})$	$O^*(2^n)$	
ISO	$O^*(2^{n/(2+\varepsilon)})$	$O^*(2^{o(n)})$	$O^*(2^n)$	
IFO	$O^*(2^{n/(2+\varepsilon)})$	$O^*(2^{o(n)})$	$\begin{cases} O^*(2^n) & \text{if } IQ_{NS} = I_{NS} \times F_{NS} \\ O^*(2^{n^2}) & \text{otherwise} \end{cases}$	
$k$ -SO	$O^*(2^{n/(2+\varepsilon)})$	$O^*(2^{o(n)})$	$O^*(2^n)$	
INSO	$O^*(2^{n/(2+\varepsilon)})$	$O^*(2^{o(n)})$	$O^*(2^n)$	

■ **Table 1** An overview of the algorithmic complexity of deciding opacity under the projection  $P: \Sigma^* \rightarrow \Gamma^*$ , where  $n$  is the number of states of the automaton and  $\varepsilon > 0$ .

(Corollary 18). This result strengthens the result of Fernau and Krebs [16] showing that if ETH is true, the universality of an  $n$ -state NFA cannot be decided in time  $O^*(2^{o(n)})$ .

## 2 Preliminaries

We assume that the reader is familiar with automata theory [22]. For a set  $S$ , the cardinality of  $S$  is denoted by  $|S|$  and the power set of  $S$  by  $2^S$ . If  $S$  is a singleton,  $S = \{x\}$ , we often simply write  $x$  instead of  $\{x\}$ . The set of all non-negative integers is denoted by  $\mathbb{N}$ .

An alphabet  $\Sigma$  is a finite nonempty set of symbols. A string over  $\Sigma$  is a finite sequence of symbols from  $\Sigma$ . The set of all strings over  $\Sigma$  is denoted by  $\Sigma^*$ ; the empty string is denoted by  $\varepsilon$ . A language  $L$  over  $\Sigma$  is a subset of  $\Sigma^*$ . For a string  $u \in \Sigma^*$ , the length of  $u$  is denoted by  $|u|$ . With every pair of alphabets  $(\Sigma, \Gamma)$  with  $\Gamma \subseteq \Sigma$ , we associate the morphism  $P: \Sigma^* \rightarrow \Gamma^*$  defined by  $P(a) = \varepsilon$ , for  $a \in \Sigma - \Gamma$ , and  $P(a) = a$ , for  $a \in \Gamma$ ; such morphisms are usually called *projections*. Intuitively, the action of the projection  $P$  is to erase all symbols that do not belong to  $\Gamma$ ; the symbols of  $\Gamma$  are usually called *observable symbols of  $\Sigma$  under the projection  $P$* . We lift the projection  $P$  from strings to languages in the usual way. The inverse projection of  $P$  is the function  $P^{-1}: \Gamma^* \rightarrow 2^{\Sigma^*}$  defined by  $P^{-1}(w) = \{w' \in \Sigma^* \mid P(w') = w\}$ .

A *nondeterministic finite automaton* (NFA) is a structure  $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ , where  $Q$  is a finite set of states,  $\Sigma$  is an input alphabet,  $I \subseteq Q$  is a set of initial states,  $F \subseteq Q$  is a set of accepting states, and  $\delta: Q \times \Sigma \rightarrow 2^Q$  is a transition function that can be extended to the domain  $2^Q \times \Sigma^*$  by induction. If the accepting states are irrelevant, we omit them and simply write  $\mathcal{A} = (Q, \Sigma, \delta, I)$ . The language accepted by  $\mathcal{A}$  from the states of  $Q_0 \subseteq Q$  by the states of  $F_0 \subseteq F$  is the set  $L_m(\mathcal{A}, Q_0, F_0) = \{w \in \Sigma^* \mid \delta(Q_0, w) \cap F_0 \neq \emptyset\}$  and the language generated by  $\mathcal{A}$  from the states of  $Q_0$  is the set  $L(\mathcal{A}, Q_0) = L_m(\mathcal{A}, Q_0, Q)$ ; in particular, the language accepted by  $\mathcal{A}$  is  $L_m(\mathcal{A}) = L_m(\mathcal{A}, I, F)$  and the language generated by  $\mathcal{A}$  is  $L(\mathcal{A}) = L(\mathcal{A}, I)$ . The NFA  $\mathcal{A}$  is *deterministic* (DFA) if  $|I| = 1$  and  $|\delta(q, a)| \leq 1$  for every state  $q \in Q$  and every symbol  $a \in \Sigma$ .

For an alphabet  $\Gamma \subseteq \Sigma$ , we define the *projected automaton* of  $\mathcal{A}$ , denoted by  $P(\mathcal{A})$ , as the reachable part of a DFA obtained from  $\mathcal{A}$  by replacing every transition  $(q, a, r)$  with the transition  $(q, P(a), r)$ , followed by the standard subset construction [22].

We define the *configuration* of  $\mathcal{A}$  as the state of the projected automaton  $P(\mathcal{A})$  of  $\mathcal{A}$ .

A (*Boolean*) *formula* consists of variables, symbols for logical connectives: conjunction, disjunction, negation; and parentheses. A *literal* is a variable or its negation. A *clause* is a disjunction of literals. A formula is in *conjunctive normal form* (CNF) if it is a conjunction of clauses. If each clause has at most  $k$  literals, the formula is in  $k$ -CNF. A formula is *satisfiable* if there is an assignment of 1 and 0 to the variables evaluating the formula to 1. Given a  $k \geq 3$  and a formula in  $k$ -CNF, the  $k$ -CNF *Boolean satisfiability problem* ( $k$ -SAT) is to decide whether the formula is satisfiable. If the formula in  $k$ -CNF has  $n$  variables, enumerating all the  $2^n$  possible truth assignments results in an  $O(2^n n^k)$ -time algorithm for  $k$ -SAT; the polynomial part  $O(n^k)$  comes from checking up to  $n^k$  clauses. We use the notation  $O^*$  to hide polynomial factors, that is,  $O^*(g(n)) = O(g(n) \cdot \text{poly}(n))$ .

The exponential time hypothesis states that 3-SAT cannot be solved in sub-exponential time  $2^{o(n)}$ , where  $n$  is the number of variables in the 3-CNF formula [23].

► **Hypothesis 1** (Exponential Time Hypothesis (ETH)). *There is some  $\varepsilon > 0$  such that 3-SAT cannot be solved in time  $O(2^{\varepsilon n})$ , where  $n$  is the number of variables in the formula.*

The strong ETH states that deciding  $k$ -SAT needs  $O^*(2^n)$  time for large  $k$  [23].

► **Hypothesis 2** (Strong ETH (SETH)). *For every  $\varepsilon > 0$ , there is some  $k \geq 3$  such that  $k$ -SAT cannot be solved in time  $O(2^{(1-\varepsilon)n})$ .*

### 3 Opacity Definitions

We now review the notions of opacity considered in this paper. We distinguish two types of opacity: those representing the secret by strings and those representing the secret by states.

Language-based opacity is a property asking whether for every secret behaviour, there is a non-secret behaviour that is the same under a considered projection; in this case, an intruder cannot distinguish the secret behaviour from a non-secret behaviour.

► **Definition 3.** *An NFA  $\mathcal{A} = (Q, \Sigma, \delta, I)$  is language-based opaque (LBO) with respect to disjoint languages  $L_S, L_{NS} \subseteq L(\mathcal{A})$ , called secret and non-secret languages, respectively, and a projection  $P: \Sigma^* \rightarrow \Gamma^*$  for  $\Gamma \subseteq \Sigma$ , if  $L_S \subseteq P^{-1}P(L_{NS})$ .*

*The LBO problem is to decide whether  $\mathcal{A}$  is LBO with respect to  $L_S$ ,  $L_{NS}$ , and  $P$ .*

This definition is general enough to capture other notions, such as *strong nondeterministic non-interference* or *non-deducibility on composition* of Best et al. [8] and Busi and Gorrieri [12], or *trace opacity* of Bryans et al. [9]. The secret and non-secret languages are often considered to be regular to ensure that the inclusion problem is decidable [2].

State-based opacity hides the secret information into states. In this paper, we consider five notions of state-based opacity. Current-state opacity requires that an intruder cannot identify, at any instance of time, whether the system is currently in a secret state.

► **Definition 4.** *An NFA  $\mathcal{A} = (Q, \Sigma, \delta, I)$  is current-state opaque (CSO) with respect to two disjoint sets  $Q_S, Q_{NS} \subseteq Q$  of secret and non-secret states, respectively, and a projection  $P: \Sigma^* \rightarrow \Gamma^*$  for  $\Gamma \subseteq \Sigma$ , if for every string  $w \in \Sigma^*$  such that  $\delta(I, w) \cap Q_S \neq \emptyset$ , there exists a string  $w' \in \Sigma^*$  such that  $P(w) = P(w')$  and  $\delta(I, w') \cap Q_{NS} \neq \emptyset$ .*

*The CSO problem is to decide whether  $\mathcal{A}$  is CSO with respect to  $Q_S$ ,  $Q_{NS}$ , and  $P$ .*

Initial-state opacity requires that an intruder can never ascertain whether the computation started in a secret state.

► **Definition 5.** An NFA  $\mathcal{A} = (Q, \Sigma, \delta, I)$  is initial-state opaque (ISO) with respect to two disjoint sets  $I_S, I_{NS} \subseteq I$  of secret and non-secret initial states, respectively, and a projection  $P: \Sigma^* \rightarrow \Gamma^*$  for  $\Gamma \subseteq \Sigma$ , if for every  $w \in L(\mathcal{A}, I_S)$ , there exists  $w' \in L(\mathcal{A}, I_{NS})$  such that  $P(w) = P(w')$ .

The ISO problem is to decide whether  $\mathcal{A}$  is ISO with respect to  $I_S, I_{NS}$ , and  $P$ .

Initial-and-final-state opacity [43] generalizes both CSO and ISO. The secret is represented as a set of pairs of an initial state and of an accepting state. Therefore, ISO is a special case of initial-and-final-state opacity where the accepting states do not play a role, while CSO is a special case where the initial states do not play a role.

► **Definition 6.** An NFA  $\mathcal{A} = (Q, \Sigma, \delta, I, F)$  is initial-and-final-state opaque (IFO) with respect to two disjoint sets  $IQ_S, IQ_{NS} \subseteq I \times F$  of secret and non-secret pairs of states, respectively, and a projection  $P: \Sigma^* \rightarrow \Gamma^*$  for  $\Gamma \subseteq \Sigma$ , if for every secret pair  $(q_0, q_f) \in IQ_S$  and every string  $w \in L_m(\mathcal{A}, q_0, q_f)$ , there exists a non-secret pair  $(q'_0, q'_f) \in IQ_{NS}$  and a string  $w' \in L_m(\mathcal{A}, q'_0, q'_f)$  such that  $P(w) = P(w')$ .

The IFO problem is to decide whether  $\mathcal{A}$  is IFO with respect to  $IQ_S, IQ_{NS}$ , and  $P$ .

The algorithmic time complexity of deciding IFO is known to be  $O^*(2^{n^2})$  in general, and  $O(2^{2n})$  if  $IQ_S = I_S \times F_S$  and  $IQ_{NS} = I_{NS} \times F_{NS}$ , for some  $I_S, I_{NS} \subseteq I$  and  $F_S, F_{NS} \subseteq F$  [43]. Our complexity in Table 1 is based on the following observations.

Consider an NFA  $\mathcal{A} = (Q, \Sigma, \delta, I, F)$  and two sets  $IQ_S, IQ_{NS} \subseteq I \times F$ . The IFO property of  $\mathcal{A}$  is unchanged if all pairs  $(s, f_1), (s, f_2), \dots, (s, f_k)$  with a common left component are replaced by a single pair of the form  $(s, \{f_1, f_2, \dots, f_k\})$ . This reduces the number of pairs to be considered to  $n$ , where  $n$  is the number of states of  $\mathcal{A}$ . For every pair  $(s_i, F_i) \in I \times 2^F$ , we define the language  $L_i = L_m(\mathcal{A}, s_i, F_i)$  and the languages

$$L_S = \bigcup_{(s_i, F_i) \in IQ_S} L_i \quad \text{and} \quad L_{NS} = \bigcup_{(s_i, F_i) \in IQ_{NS}} L_i.$$

Then, deciding whether  $\mathcal{A}$  is IFO with respect to  $IQ_S, IQ_{NS}$ , and  $P$  is equivalent to deciding whether the inclusion  $P(L_S) \subseteq P(L_{NS})$  holds true. Since both  $L_S$  and  $L_{NS}$  can be represented by NFAs consisting of at most  $n$  copies of  $\mathcal{A}$ , they have  $O(n^2)$  states. The inclusion  $P(L_S) \subseteq P(L_{NS})$  of languages of two NFAs can be tested in time  $O(n^2 2^{n^2}) = O^*(2^{n^2})$ , which is a complexity upper bound that coincides with the bound of Saboori and Hadjicostis [36], who used trellis automata.

If  $IQ_{NS} = I_{NS} \times F_{NS} \subseteq I \times F$ , then the NFA for  $L_{NS}$  coincides with  $\mathcal{A}$  where the initial states are  $I_{NS}$  and the final states are  $F_{NS}$ . In particular, this automaton has  $n$  states, and therefore the inclusion  $P(L_S) \subseteq P(L_{NS})$  can be tested in time  $O(n^2 2^n) = O^*(2^n)$ .

The notion of  $k$ -step opacity generalizes CSO by requiring that the intruder cannot ascertain the secret in the current and  $k$  subsequent states. By definition, CSO is equivalent to 0-step opacity. We use a slight generalisation of a definition of Saboori and Hadjicostis [35] that was formulated by Balun and Masopust [6].

► **Definition 7.** An NFA  $\mathcal{A} = (Q, \Sigma, \delta, I)$  is  $k$ -step opaque ( $k$ -SO), for a given  $k \in \mathbb{N} \cup \{\infty\}$ , with respect to two disjoint sets  $Q_S, Q_{NS} \subseteq Q$  of secret and non-secret states, respectively, and a projection  $P: \Sigma^* \rightarrow \Gamma^*$  for  $\Gamma \subseteq \Sigma$ , if for every string  $st \in L(\mathcal{A})$  such that  $|P(t)| \leq k$  and  $\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$ , there exists a string  $s't' \in L(\mathcal{A})$  such that  $P(s) = P(s')$ ,  $P(t) = P(t')$ , and  $\delta(\delta(I, s') \cap Q_{NS}, t') \neq \emptyset$ .

The  $k$ -SO problem is to decide whether  $\mathcal{A}$  is  $k$ -SO with respect to  $Q_S, Q_{NS}$ , and  $P$ .

A special case of  $k$ -SO for  $k$  being infinity is called infinite-step opacity (INSO). These two notions are closely related for finite automata, because an  $n$ -state automaton is infinite-step opaque if and only if it is  $(2^n - 2)$ -step opaque [45].

For the state-based opacity notions, we may assume, without loss of generality, that the projection  $P$  is an identity; indeed, the transitions labeled by symbols from  $\Sigma - \Gamma$  can be seen as  $\varepsilon$ -transitions, and hence they can be removed by the classical algorithm eliminating  $\varepsilon$ -transitions [22]. This algorithm does not change the number of states but can quadratically increase the number of transitions. In the sequel, we omit the projection if it is an identity.

#### 4 Lower Bounds under Strong Exponential Time Hypothesis

We now show that under the strong exponential time hypothesis, there is no algorithm deciding current-state opacity that would be significantly faster than the best known algorithm.

► **Theorem 8.** *Unless SETH fails, there is no algorithm deciding whether a given  $n$ -state NFA is CSO that runs in time  $O^*(2^{n/(2+\varepsilon)})$ , for any  $\varepsilon > 0$ .*

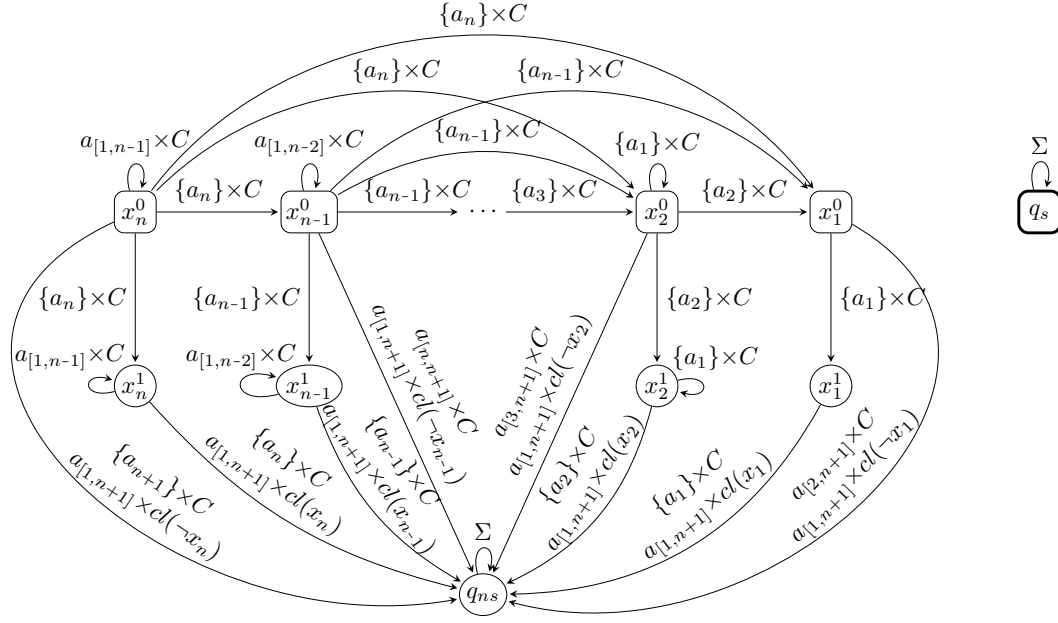
**Proof.** For a given formula  $\varphi$  in  $k$ -CNF with  $n$  variables  $X = \{x_1, \dots, x_n\}$  and  $m$  clauses  $C = \{c_1, \dots, c_m\}$ , we construct, in polynomial time, an instance of CSO consisting of an NFA  $\mathcal{A}_\varphi$  with  $N = 2n + 2$  states and of sets of secret and non-secret states  $Q_S$  and  $Q_{NS}$ , respectively, such that  $\mathcal{A}_\varphi$  is CSO with respect to  $Q_S$  and  $Q_{NS}$  if and only if  $\varphi$  is satisfiable. As a result, if there was an algorithm solving CSO in time  $O^*(2^{N/(2+\varepsilon)})$ , then there would be an algorithm solving  $k$ -SAT in time  $O(\text{poly}(n)) + O^*(2^{(2n+2)/(2+\varepsilon)}) = O^*(2^{(1-\delta)n})$ , for  $\delta = \varepsilon/(2 + \varepsilon) > 0$ , which contradicts SETH, and proves the theorem.

Intuitively, we construct the NFA  $\mathcal{A}_\varphi$  such that when  $A_\varphi$  reads a string of a particular type (based on Zimin words), it is forced to examine all possible assignments to the variables  $x_1, \dots, x_n$ . If none of the assignments satisfies  $\varphi$ , then, after reading the whole string, the automaton  $A_\varphi$  ends up in a configuration that contains only secret states, rendering thus  $A_\varphi$  not CSO. On the other hand, when a satisfying assignment is encountered (or the string is not of the particular type), a non-secret state is permanently added to the configuration of  $A_\varphi$ , and hence  $A_\varphi$  is CSO.

Formally, the NFA  $\mathcal{A}_\varphi = (Q, \Sigma, \delta, I)$ , where the set of states is  $Q = \{q_s, q_{ns}\} \cup \{x_i^0, x_i^1 \mid x_i \in X\}$  with  $x_i^r$  representing the assignment of  $r \in \{0, 1\}$  to the variable  $x_i$ , the alphabet  $\Sigma = \Gamma = \{a_1, \dots, a_{n+1}\} \times C$ , that is, the projection  $P$  is an identity, and the set of secret states is  $Q_S = \{q_s\}$ , that is, the state  $q_s$  is the only secret state, the remaining states are non-secret. For an illustration of the construction, the reader may follow Example 14 together with the rest of the proof.

Let  $L$  be the set of literals of  $\varphi$ . We use the function  $cl: L \rightarrow 2^C$  that assigns to a literal  $\ell$  the set  $cl(\ell) = \{c \in C \mid \ell \in c\}$  of clauses containing  $\ell$ , and define the transition function  $\delta$  as follows, see Figure 1 for an illustration:

- The self-loops  $(q_s, \sigma, q_s)$  and  $(q_{ns}, \sigma, q_{ns})$  belong to  $\delta$  for every  $\sigma \in \Sigma$ ;
- For every state  $x_i^0$  and every  $c \in C$ ,
  - the transition  $(x_i^0, (a_i, c), x_i^1) \in \delta$ ;
  - the self-loop  $(x_i^0, (a_j, c), x_i^0) \in \delta$  for  $1 \leq j \leq i - 1$ ;
  - the transition  $(x_i^0, (a_i, c), x_j^0) \in \delta$  for  $1 \leq j \leq i - 1$ ;
  - the transition  $(x_i^0, (a_j, c), q_{ns}) \in \delta$  for  $i + 1 \leq j \leq n + 1$ ;
  - the transition  $(x_i^0, (a_j, c), q_{ns}) \in \delta$  for  $1 \leq j \leq n + 1$  and  $c \in cl(\neg x_i)$ ;
- For every state  $x_i^1$  and  $c \in C$ ,
  - the transition  $(x_i^1, (a_i, c), q_{ns}) \in \delta$ ;



■ **Figure 1** The NFA  $A_\varphi$  of Theorem 8, where the initial states are squared, the single secret state  $q_s$  is in bold, and for positive integers  $i \leq j$ ,  $[i, j] = \{i, i + 1, \dots, j\}$  and  $a_{[i, j]} = \{a_r \mid r \in [i, j]\}$ .

- the self-loop  $(x_i^1, (a_j, c), x_i^1) \in \delta$ , for  $1 \leq j \leq i - 1$ ;
- the transition  $(x_i^1, (a_j, c), q_{ns}) \in \delta$  for  $1 \leq j \leq n + 1$  and  $c \in cl(x_i)$ .

Finally, the set of initial states is  $I = \{q_s\} \cup \{x_1^0, \dots, x_n^0\}$ , corresponding to the assignment of 0 to all variables of  $\varphi$ .

We now define a language  $W_\varphi = Z_n \cdot (\{a_{n+1}\} \times C)$ , where  $Z_n$  is a language over the alphabet  $\{a_1, a_2, \dots, a_n\} \times C$  recursively defined as follows:

$$Z_1 = \{a_1\} \times C \quad \text{and} \quad Z_i = Z_{i-1} \cdot (\{a_i\} \times C) \cdot Z_{i-1}, \quad \text{for } 1 < i \leq n.$$

Such strings are known as *Zimin words* and it is well known that any string of  $Z_n$  is of length  $2^n - 1$  [40] and that

$$\text{the symbol on the } \ell\text{th position of any string from } Z_n \text{ is of the form } \{a_j\} \times C, \text{ where } (1) \\ j - 1 \text{ is the number of trailing zeros in the binary representation of } \ell \text{ [41].}$$

We finish the proof in a series of claims. The first claim shows that along any string of  $Z_n$ , the states  $\{x_i^0, x_i^1 \mid x_i \in X\}$  of  $\mathcal{A}_\varphi$  encode all possible assignments to the variables.

▷ **Claim 9.** Let  $\mathcal{A}_\varphi^X$  denote  $\mathcal{A}_\varphi$  without the states  $q_s$  and  $q_{ns}$  and the corresponding transitions. For every  $w \in Z_n$ , after reading the prefix of  $w$  of length  $\ell \leq 2^n - 1$ , the configuration of  $\mathcal{A}_\varphi^X$  is  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_1^{r_1}\}$ , where  $r_n r_{n-1} \dots r_1$  represents  $\ell$  in binary.

▷ **Claim 10.** Every configuration of  $\mathcal{A}_\varphi$  contains the secret state  $q_s$ .

By Claims 9 and 10, and because only the state  $q_s$  itself is reachable from  $q_s$  and only the state  $q_{ns}$  itself is reachable from  $q_{ns}$ , we have the following observation specifying the computation of  $\mathcal{A}_\varphi$  along the strings of  $Z_n$ .

▷ **Claim 11.** After reading the prefix of  $w \in Z_n$  of length  $\ell \leq 2^n - 1$ , the configuration of  $\mathcal{A}_\varphi$  is either  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_1^{r_1}\} \cup \{q_s\}$  or  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_1^{r_1}\} \cup \{q_s, q_{ns}\}$ , where  $r_n r_{n-1} \dots r_1$  represents  $\ell$  in binary. ◁

If there is a satisfying assignment, then the non-secret state  $q_{ns}$  is reached by  $\mathcal{A}_\varphi$ .

▷ **Claim 12.** For every prefix  $w$  of a string in  $Z_n$ , if the configuration of  $\mathcal{A}_\varphi$  after reading  $w$  satisfies  $\varphi$ , then after reading any further symbol in  $\Sigma$ , the configuration of  $\mathcal{A}_\varphi$  contains  $q_{ns}$ .

We now show that if  $\varphi$  is satisfiable, then  $\mathcal{A}_\varphi$  is CSO. To this end, we consider an arbitrary string  $w$  over  $\Sigma$ , and we denote by  $u$  the longest prefix of  $w$  that is a prefix of a string in  $Z_n$ . Let  $\ell$  be the minimal number such that its binary representation  $r_n r_{n-1} \cdots r_1$  is a satisfying assignment to the variables of  $\varphi$ .

If  $\ell \leq |u|$ , then, by Claim 11, the configuration of  $\mathcal{A}_\varphi$  after reading any prefix of  $u$  of length  $\ell' \leq \ell$  contains non-secret states  $x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_1^{r_1}$ , where  $r_n r_{n-1} \cdots r_1$  represents  $\ell'$  in binary, and, by Claim 12, the  $(\ell + 1)$ st symbol of  $w$  moves  $\mathcal{A}_\varphi$  to a configuration that contains  $q_{ns}$ ; that is,  $\mathcal{A}_\varphi$  is CSO.

If  $\ell > |u|$ , we have  $w = u(a_s, c)v$  for  $(a_s, c) \in \Sigma$  and  $v \in \Sigma^*$ . Because  $\varphi$  is satisfiable, we have  $|u| < 2^n - 1$ . By Claim 11, the configuration of  $\mathcal{A}_\varphi$  after reading  $u$  contains  $x_n^{r_n}, \dots, x_1^{r_1}$ , where  $r_n r_{n-1} \cdots r_1$  represents  $|u|$  in binary. Let  $r_t$  be the rightmost zero of  $r_n r_{n-1} \cdots r_1$ , that is,  $r_n r_{n-1} \cdots r_1 = r_n r_{n-1} \cdots r_{t+1} 0 1 \cdots 1$ . Then,  $|u| + 1$  is  $r_n r_{n-1} \cdots r_{t+1} 1 0 \cdots 0$  in binary and, by (1), the symbol  $(a_s, c) \notin \{a_t\} \times C$ . However, if  $s < t$ , then  $x_s^1$  goes to state  $q_{ns}$  under  $\{a_s\} \times C$ , while if  $s > t$ , then  $x_t^0$  goes to state  $q_{ns}$  under  $\{a_s\} \times C$ . In both cases, the non-secret state  $q_{ns}$  is in the next configuration of  $\mathcal{A}_\varphi$ , and hence  $\mathcal{A}_\varphi$  is CSO.

To prove that if  $\varphi$  is not satisfiable, then  $\mathcal{A}_\varphi$  is not CSO, we use the following claim.

▷ **Claim 13.** If  $\varphi$  is not satisfiable, there is a string  $w_\varphi \in Z_n$  such that the configuration of  $\mathcal{A}_\varphi$  after reading  $w_\varphi$  is  $\{x_n^1, x_{n-1}^1, \dots, x_1^1\} \cup \{q_s\}$ .

We now show that if  $\varphi$  is not satisfiable, then  $\mathcal{A}_\varphi$  is not CSO. To this end, we consider the string  $w_\varphi$  constructed in Claim 13, which we extend to a string from  $W_\varphi = Z_n \cdot (\{a_{n+1}\} \times C)$  by adding a symbol of the form  $\{a_{n+1}\} \times C$ . Since  $\varphi$  is not satisfiable, there is a clause  $c \in C$  that is not satisfied by the assignment of 1 to the variables; that is, there is  $c \notin \bigcup_{i=1}^n cl(x_i)$ . Then, the string  $w_\varphi(a_{n+1}, c)$  moves the automaton  $\mathcal{A}_\varphi$  from the configuration  $\{x_n^1, x_{n-1}^1, \dots, x_1^1\} \cup \{q_s\}$  to the configuration  $\{q_s\}$ , and hence  $\mathcal{A}_\varphi$  is not CSO. ◀

We now illustrate the construction.

► **Example 14.** For simplicity, we consider a 2-CNF formula

$$\varphi = (x_2 \vee x_2) \wedge (x_1 \vee x_2) \wedge (\neg x_1 \vee x_3) \wedge (\neg x_2 \vee \neg x_3) \wedge (x_3)$$

with three variables  $x_1, x_2, x_3$  and five clauses  $c_1 = \{x_2, x_2\}$ ,  $c_2 = \{x_1, x_2\}$ ,  $c_3 = \{\neg x_1, x_3\}$ ,  $c_4 = \{\neg x_2, \neg x_3\}$ , and  $c_5 = \{x_3\}$ . The automaton  $\mathcal{A}_\varphi = (Q, \Sigma, \delta, \{q_s, x_1^0, x_2^0, x_3^0\})$  is depicted in Figure 2, where  $Q = \{q_s, q_{ns}\} \cup \{x_1^0, x_1^1, x_2^0, x_2^1, x_3^0, x_3^1\}$ ,  $\Sigma = \Gamma = \{a_1, a_2, a_3, a_4\} \times \{c_1, c_2, c_3, c_4, c_5\}$ , and  $q_s$  is the only secret state. Since  $\varphi$  is not satisfiable,  $\mathcal{A}_\varphi$  is not CSO; indeed, the string  $w = (a_1, c_1)(a_2, c_2)(a_1, c_5)(a_3, c_3)(a_1, c_1)(a_2, c_1)(a_1, c_4)(a_4, c_4)$  moves  $\mathcal{A}_\varphi$  to the configuration  $\{q_s\}$  consisting solely of the secret state, cf. Figure 3 depicting the reachable configurations of  $\mathcal{A}_\varphi$ .

On the other hand, if we consider the formula  $\varphi' = c_1 \wedge c_2 \wedge c_3 \wedge c_4$ , then  $\varphi'$  is satisfiable, and hence the NFA  $\mathcal{A}_{\varphi'}$  obtained from  $\mathcal{A}_\varphi$  by removing all transitions under symbols containing  $c_5$ , is CSO; it is visible from the reachable configurations of  $\mathcal{A}_{\varphi'}$  depicted in Figure 4.

The considered problems are all PSPACE-complete, and hence reducible to each other in polynomial time. However, this fact does not provide us with much information about the reductions. Even though some particular reductions have been discussed in the literature by Wu and Lafortune [43] and Balun and Masopust [6, 7], they are in most cases not suitable to prove lower bounds.



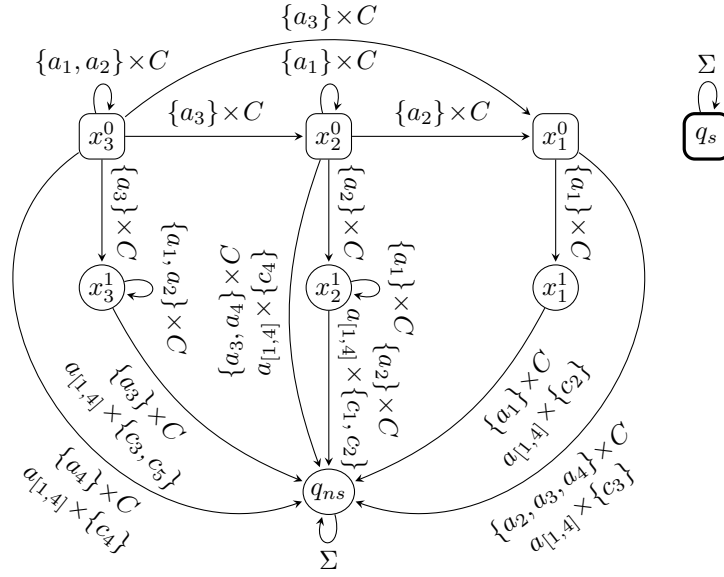


Figure 2 The NFA  $A_\varphi$  illustrating Theorem 8; the initial states are squared.

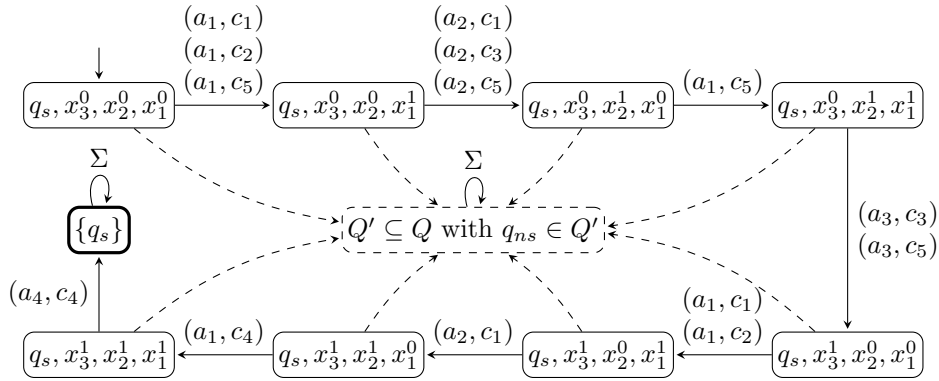


Figure 3 The configurations of  $A_\varphi$ —all undefined transitions go to the dashed middle state.

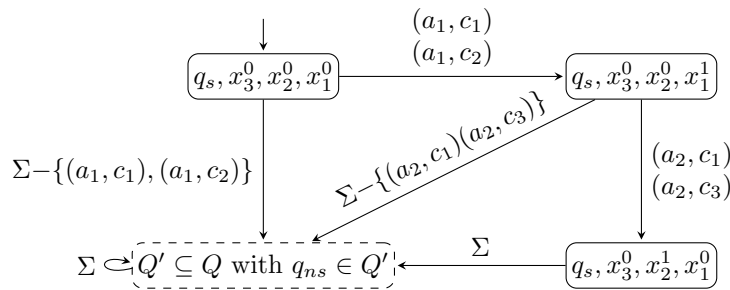


Figure 4 The configurations of  $A_{\varphi'}$ .

We now discuss the case of other types of opacity.

► **Corollary 15.** *Unless SETH fails, there is no algorithm deciding if a given  $n$ -state NFA is LBO/ISO/IFO/ $k$ -SO/INSO that runs in time  $O^*(2^{n/(2+\varepsilon)})$ , for any  $\varepsilon > 0$ .*

**Proof.** Consider the instance of CSO given by the NFA  $\mathcal{A}$  and the sets  $Q_S$  and  $Q_{NS}$  constructed in the proof of Theorem 8. Then,  $\mathcal{A}$  is CSO with respect to  $Q_S = \{q_s\}$  and  $Q_{NS} = Q - Q_S$  if and only if  $\mathcal{A}$  is LBO with respect to  $L_S = L_m(\mathcal{A}, q_s, q_s) = \Sigma^*$  and  $L_{NS} = L_m(\mathcal{A}, \{x_1^0, \dots, x_n^0\}, Q - \{q_s\})$ . Since the parts of  $\mathcal{A}$  corresponding to languages  $L_S$  and  $L_{NS}$  are disjoint, the instance of LBO may be encoded directly into  $\mathcal{A}$  by defining the corresponding states accepting the languages  $L_S$  and  $L_{NS}$ . Hence, the instance of LBO is of the same size as the instance of CSO. Therefore, if we solved the instance of LBO in time  $O^*(2^{(1-\delta)n})$ , we would also solve the instance of CSO in time  $O^*(2^{(1-\delta)n})$ .

Since there is no transition in  $\mathcal{A}$  from the sole secret state  $q_s$  to another state, the NFA  $\mathcal{A}$  is CSO if and only if  $\mathcal{A}$  is  $k$ -SO, for any  $k \in \mathbb{N} \cup \{\infty\}$ , and hence the result holds for  $k$ -SO as well as for INSO.

Furthermore, the NFA  $\mathcal{A}$  is CSO with respect to the sets  $Q_S$  and  $Q_{NS}$  if and only if  $\mathcal{A}$  is ISO with respect to the secret initial state  $I_S = \{q_s\}$  and non-secret initial states  $I_{NS} = \{x_1^0, \dots, x_n^0\}$ . Indeed, since  $L(\mathcal{A}, I_S) = \Sigma^*$ , the NFA  $\mathcal{A}$  is not CSO if and only if there is a string  $w \in \Sigma^*$  that moves  $\mathcal{A}$  from the initial configuration  $I_{NS}$  to the configuration  $\emptyset$ , which is if and only if  $\mathcal{A}$  is not ISO. As a result, solving ISO in time  $O^*(2^{(1-\delta)n})$  would solve CSO in time  $O^*(2^{(1-\delta)n})$ .

Finally, if all states of  $\mathcal{A}$  are accepting, then  $\mathcal{A}$  is ISO with respect to  $I_S = \{q_s\}$  and  $I_{NS} = \{x_1^0, \dots, x_n^0\}$  if and only if  $\mathcal{A}$  is IFO with respect to  $IQ_S = \{(q_s, q_s)\}$  and  $IQ_{NS} = I_{NS} \times Q$ ; hence, solving IFO in time  $O^*(2^{(1-\delta)n})$  would solve ISO in time  $O^*(2^{(1-\delta)n})$ . ◀

## 5 Lower Bounds under Exponential Time Hypothesis

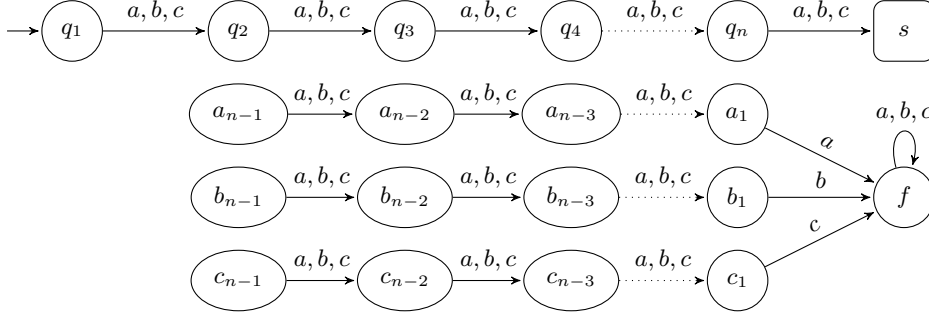
The number of symbols in the NFA constructed in Theorem 8 depends on the number of clauses in the instance of SAT. Since the standard binary encoding of symbols does not work under SETH, it is an open problem whether the results of Theorem 8 and Corollary 15 also hold for a fixed-sized alphabet.

Although we do not answer this question, we provide a lower bound for NFAs over a binary alphabet under ETH. Namely, we show that there is no algorithm solving the considered notions of opacity for such  $n$ -state NFAs that runs in time  $O^*(2^{o(n)})$ . We obtain the result by adjusting the construction of Fernau and Krebs [16], who showed that there is no algorithm solving the universality problem for  $n$ -state NFAs over a binary alphabet that runs in time  $O^*(2^{o(n)})$  unless ETH fails, and by using the observation of Cassez et al. [13] that universality can be reduced to opacity.

► **Theorem 16.** *Unless ETH fails, there is no algorithm deciding whether a given  $n$ -state NFA (over a binary alphabet) is CSO that runs in time  $O^*(2^{o(n)})$ .*

**Proof.** A 3-coloring of a graph  $G = (V, E)$  is a function  $\mu: V \rightarrow \{a, b, c\}$ . The coloring is proper if  $\mu(u) \neq \mu(v)$  whenever  $uv \in E$ . The 3-COLORING problem is to decide, given a graph  $G$ , whether there is a proper 3-coloring of  $G$ .

For a graph  $G$  with  $n$  vertices,  $V = \{v_1, v_2, \dots, v_n\}$ , and  $m$  edges, we construct an NFA  $\mathcal{A} = (Q, \Sigma, \delta, I)$ , where the states are  $Q = \{s, f\} \cup \{q_1, \dots, q_n\} \cup \{x_1, \dots, x_{n-1} \mid x \in \{a, b, c\}\}$ , the alphabet is  $\Sigma = \Gamma = \{a, b, c\}$ , the initial state is  $I = \{q_1\}$ , the secret state is  $Q_S = \{s\}$ , and the non-secret states are  $Q_{NS} = Q - Q_S$ . We define the transition function  $\delta$  as shown



■ **Figure 5** The main part of the NFA  $\mathcal{A}$  resulting from the reduction of Theorem 16 without the transitions corresponding to the edges of  $G$ . The secret state is state  $s$ .

in Figure 5, and further extended it by adding three transitions  $(q_i, a, a_{j-i})$ ,  $(q_i, b, b_{j-i})$ , and  $(q_i, c, c_{j-i})$  for every edge  $v_i v_j \in E$  with  $i < j$ .

Intuitively, the coloring of  $G$  is encoded as a string  $w = c_1 \cdots c_n$  of length  $n$ , where  $c_i$  is the color of vertex  $i$ , and  $\mathcal{A}$  is CSO with respect to  $\{s\}$  and  $Q - \{s\}$  if and only if the non-secret state  $f$  is reachable under  $w$ ; indeed, the secret state  $s$  is reachable under every string of length  $n$ .

Fernau and Krebs [16] showed that  $\mathcal{A}$  ends up in state  $f$  under  $w$  if and only if  $w$  encodes a coloring that is not proper. Therefore, if  $w$  is a proper 3-coloring of  $G$ , then  $w$  does not move  $\mathcal{A}$  to state  $f$ ; that is, only the secret state  $s$  is reached under  $w$ , and hence  $\mathcal{A}$  is not CSO with respect to  $\{s\}$  and  $Q - \{s\}$ . On the other hand, if  $G$  does not have a proper 3-coloring, then every string of length  $n$  moves  $\mathcal{A}$  to both secret state  $s$  and non-secret state  $f$ ; that is,  $\mathcal{A}$  is CSO with respect to  $\{s\}$  and  $Q - \{s\}$ .

If  $G$  has  $n$  vertices and  $m$  edges, then  $\mathcal{A}$  has  $N = 4n - 1$  states and  $M = 12n + 3m - 12$  transitions. If there was an  $O^*(2^{o(N)})$ -time algorithm deciding CSO, we could reduce the instance of 3-COLORING to an instance of CSO in time  $O(N + M)$ , and solve CSO in time  $O^*(2^{o(N)})$ . Altogether, we could solve 3-COLORING in time  $O(N + M) + O^*(2^{o(N)}) = O^*(2^{o(n)})$ , which contradicts ETH. ◀

Similarly to the discussion in the previous section, to prove the lower bound for the other notions of opacity, it seems natural to combine the construction of the previous proof with the existing reductions among the notions [6, 7, 43]. However, most of the reductions result in too large, though polynomial, instances, and hence they are not suitable for our purposes. Therefore, new reductions are needed.

► **Corollary 17.** *Unless ETH fails, there is no algorithm deciding if a given  $n$ -state NFA (over a binary alphabet) is LBO/ISO/IFO/ $k$ -SO/INSO that runs in time  $O^*(2^{o(n)})$ .*

**Proof.** For the NFA  $\mathcal{A}$  of Theorem 16, we have  $\mathcal{A}$  is CSO with respect to  $\{s\}$  and  $Q - \{s\}$  if and only if  $\mathcal{A}$  is LBO with respect to  $L_S = L_m(\mathcal{A}, q_1, s)$  and  $L_{NS} = L_m(\mathcal{A}, q_1, Q - \{s\})$  [43]. If we could solve LBO in time  $O^*(2^{o(n)})$ , we could solve 3-COLORING in time  $O^*(2^{o(n)})$ .

Furthermore, since there is no transition from the sole secret state  $s$ , the automaton  $\mathcal{A}$  is CSO with respect to  $\{s\}$  and  $Q - \{s\}$  if and only if  $\mathcal{A}$  is  $k$ -SO with respect to  $\{s\}$  and  $Q - \{s\}$ , for any  $k \in \mathbb{N} \cup \{\infty\}$ . Therefore, the result holds for  $k$ -SO as well as for INSO.

Now, we take the NFA  $\mathcal{A}$  and add a copy of states  $q_1, q_2, \dots, q_n$ , denoted by  $q'_1, q'_2, \dots, q'_n$ , together with all transitions to states different from  $s$ , that is, we add  $(q'_i, x, p)$  for every transition  $(q_i, x, p)$  with  $p \neq s$ . We set the states  $q_1$  and  $q'_1$  initial, and denote the result by  $\mathcal{A}'$ . Then, the NFA  $\mathcal{A}$  is CSO with respect to  $\{s\}$  and  $Q - \{s\}$  if and only if  $\mathcal{A}'$  is ISO with

respect to  $I_S = \{q_1\}$  and  $I_{NS} = \{q'_1\}$ . Indeed, if  $\mathcal{A}$  is CSO with respect to  $\{s\}$  and  $Q - \{s\}$ , then for every  $w$  moving  $\mathcal{A}$  to state  $s$ , there is  $w'$  moving  $\mathcal{A}$  to state  $f$ ; and so do the strings  $w$  and  $w'$  in  $\mathcal{A}'$ , which shows that  $\mathcal{A}'$  is ISO with respect to  $I_S = \{q_1\}$  and  $I_{NS} = \{q'_1\}$ . On the other hand, if  $\mathcal{A}$  is not CSO with respect to  $\{s\}$  and  $Q - \{s\}$ , then there is  $w$  moving  $\mathcal{A}$  only to state  $s$ , and hence  $w$  cannot be read by  $\mathcal{A}'$  from state  $q'_1$ , which shows that  $\mathcal{A}'$  is not ISO with respect to  $I_S = \{q_1\}$  and  $I_{NS} = \{q'_1\}$ . If we could solve ISO in time  $O^*(2^{o(n)})$ , we could solve 3-COLORING in time  $O^*(2^{o(n)})$  by reducing it to ISO and solving ISO in time  $O^*(2^{o(N+n)}) = O^*(2^{o(n)})$ , for  $N = 4n - 1$ .

If we in addition set the states  $s$  and  $f$  accepting, then  $\mathcal{A}'$  is ISO with respect to  $I_S = \{q_1\}$  and  $I_{NS} = \{q'_1\}$  if and only if  $\mathcal{A}'$  is IFO with respect to  $IQ_S = \{(q_1, s)\}$  and  $IQ_{NS} = \{(q'_1, f)\}$ , and hence if we could solve IFO in time  $O^*(2^{o(n)})$ , we could solve 3-COLORING in time  $O^*(2^{o(n)})$ . ◀

## 6 Discussion and Conclusions

We showed that if the strong exponential time hypothesis holds true, then, for any  $c > 2$ , there are no algorithms deciding various types of opacity in time  $O^*(2^{n/c})$ . Therefore, the current algorithms cannot be significantly improved.

More precisely, the results say that there are no algorithms deciding various types of opacity in time  $O^*(\sqrt{2}^n) = O^*(1.414213562^n)$ . However, the results admit the existence of algorithms deciding opacity in time  $O^*(1.5^n)$ . Whether such algorithms exist or whether the current lower bounds can be strengthened remains an open problem.

The construction used in the proof of Theorem 8 can be utilized to improve the conditional lower bound of deciding universality for NFAs. The universality problem for NFAs asks whether, given an NFA, the NFA accepts all strings over its alphabet. If we set the only secret state  $q_s$  of the NFA  $\mathcal{A}$  of Theorem 8 to be non-accepting and all the other states to be accepting, we obtain an NFA that is universal if and only if the automaton  $\mathcal{A}$  is CSO with respect to  $\{q_s\}$  and  $Q - \{q_s\}$ . We thus have the following consequence improving the result of Fernau and Krebs [16].

► **Corollary 18.** *Unless SETH fails, there is no algorithm deciding whether a given  $n$ -state NFA is universal that runs in time  $O^*(2^{n/(2+\varepsilon)})$ , for any  $\varepsilon > 0$ .* ◀

Consequently, we immediately have the following result.

► **Corollary 19.** *Given two NFAs  $\mathcal{A}_1$  and  $\mathcal{A}_2$  with  $n_1$  and  $n_2$  states, respectively, let  $n = \max(n_1, n_2)$ . Unless SETH fails, there is no algorithm deciding whether  $L_m(\mathcal{A}_1) \subseteq L_m(\mathcal{A}_2)$  in time  $O^*(2^{n/(2+\varepsilon)})$ , and there is no algorithm deciding whether  $L_m(\mathcal{A}_1) = L_m(\mathcal{A}_2)$  in time  $O^*(2^{n/(2+\varepsilon)})$ , for any  $\varepsilon > 0$ .* ◀

We left the question whether Theorem 8 also holds for NFAs over a fixed-size alphabet open. Although we did not answer this question, we showed that ETH implies the non-existence of sub-exponential-time algorithms deciding various types of opacity over a binary alphabet.

Inspecting Table 1, the reader may notice quite a large gap between the lower and upper bounds for the verification of IFO without any restrictions on the form of non-secret pairs. To improve the upper bound or to (conditionally) show that no such improvements are possible is a challenging open problem.

It is worth noticing that the construction in the proof of Corollary 15 produces an instance of a special case of the problem where the non-secret pairs are of the form  $IQ_{NS} = I_{NS} \times F_{NS}$ , and hence the special case is tight under the strong exponential time hypothesis.

---

**References**

---

- 1 Rajeev Alur, Pavol Černý, and Steve Zdancewic. Preserving secrecy under refinement. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *International Colloquium on Automata, Languages and Programming, ICALP 2006, Venice, Italy, July 10-14, 2006*, volume 4052 of *Lecture Notes in Computer Science*, pages 107–118. Springer, 2006. doi:10.1007/11787006\10.
- 2 Peter R. J. Asveld and Anton Nijholt. The inclusion problem for some subclasses of context-free languages. *Theoretical Computer Science*, 230(1-2):247–256, 2000. doi:10.1016/S0304-3975(99)00113-9.
- 3 Éric Badouel, Marek A. Bednarczyk, Andrzej M. Borzyszkowski, Benoît Caillaud, and Philippe Darondeau. Concurrent secrets. *Discrete Event Dynamic Systems*, 17(4):425–446, 2007. doi:10.1007/s10626-007-0020-5.
- 4 Éric Badouel, Marek A. Bednarczyk, Andrzej M. Borzyszkowski, Benoît Caillaud, and Philippe Darondeau. Concurrent secrets. *Discrete Event Dynamic Systems*, 17(4):425–446, 2007. doi:10.1007/s10626-007-0020-5.
- 5 Jiří Balun and Tomáš Masopust. On opacity verification for discrete-event systems. In *IFAC World Congress, Berlin, Germany, July 11-17, 2020*, pages 2105–2110, 2020. doi:10.1016/j.ifacol.2020.12.2524.
- 6 Jiří Balun and Tomáš Masopust. Comparing the notions of opacity for discrete-event systems. *Discrete Event Dynamic Systems*, 31(4):553–582, 2021. doi:10.1007/s10626-021-00344-2.
- 7 Jiří Balun and Tomáš Masopust. On transformations among opacity notions. In *IEEE International Conference on Systems, Man, and Cybernetics, SMC 2022, Prague, Czech Republic, October 9-12, 2022*, pages 3012–3017. IEEE, 2022. doi:10.1109/SMC53654.2022.9945608.
- 8 Eike Best, Philippe Darondeau, and Roberto Gorrieri. On the decidability of non interference over unbounded Petri nets. In Konstantinos Chatzikokolakis and Véronique Cortier, editors, *International Workshop on Security Issues in Concurrency, SecCo 2010, Paris, France, August 30, 2010*, volume 51 of *EPTCS*, pages 16–33, 2010. doi:10.4204/EPTCS.51.2.
- 9 Jeremy W. Bryans, Maciej Koutny, Laurent Mazaré, and Peter Y. A. Ryan. Opacity generalised to transition systems. *International Journal of Information Security*, 7(6):421–435, 2008. doi:10.1007/s10207-008-0058-x.
- 10 Jeremy W. Bryans, Maciej Koutny, and Peter Y. A. Ryan. Modelling dynamic opacity using petri nets with silent actions. In Theodosios Dimitrakos and Fabio Martinelli, editors, *Formal Aspects in Security and Trust: Second IFIP TC1 WG1.7 Workshop on Formal Aspects in Security and Trust (FAST), an event of the 18th IFIP World Computer Congress, Toulouse, France, August 22-27, 2004*, volume 173 of *IFIP*, pages 159–172. Springer, 2004. doi:10.1007/0-387-24098-5\12.
- 11 Jeremy W. Bryans, Maciej Koutny, and Peter Y.A. Ryan. Modelling opacity using Petri nets. *Electronic Notes in Theoretical Computer Science*, 121:101–115, 2005. Proceedings of the 2nd International Workshop on Security Issues with Petri Nets and other Computational Models (WISP 2004). doi:10.1016/j.entcs.2004.10.010.
- 12 Nadia Busi and Roberto Gorrieri. Structural non-interference in elementary and trace nets. *Mathematical Structures in Computer Science*, 19(6):1065–1090, 2009. doi:10.1017/S0960129509990120.
- 13 Franck Cassez, Jérémy Dubreil, and Hervé Marchand. Synthesis of opaque systems with static and dynamic masks. *Formal Methods in System Design*, 40(1):88–115, 2012. doi:10.1007/s10703-012-0141-9.
- 14 Roger Dingledine, Nick Mathewson, and Paul Syverson. Reputation in P2P anonymity systems. In *Workshop on Economics of Peer-to-Peer Systems, P2P Econ 2003, Berkeley, CA, USA, June 5-6, 2003*, 2003.

- 15 Jeremy Dubreil, Philippe Darondeau, and Herve Marchand. Opacity enforcing control synthesis. In *International Workshop on Discrete Event Systems, WODES 2008, Gothenburg, Sweden, May 28-30, 2008*, pages 28–35. IEEE, 2008. doi:10.1109/WODES.2008.4605918.
- 16 Henning Fernau and Andreas Krebs. Problems on finite automata and the exponential time hypothesis. *Algorithms*, 10(1):24, 2017. doi:10.3390/a10010024.
- 17 R. Focardi and R. Gorrieri. A taxonomy of trace-based security properties for CCS. In *Computer Security Foundations Workshop VII, Franconia, NH, USA, June 14-16, 1994*, pages 126–136, 1994. doi:10.1109/CSFW.1994.315941.
- 18 Rômulo Meira Góes, Blake C. Rawlings, Nicholas Recker, Gregory Willett, and Stéphane Lafortune. Demonstration of indoor location privacy enforcement using obfuscation. *IFAC-PapersOnLine*, 51(7):145–151, 2018. doi:10.1016/j.ifacol.2018.06.293.
- 19 N. B. Hadj-Alouane, S. Lafrance, Feng Lin, J. Mullins, and M. M. Yeddes. On the verification of intransitive noninterference in multilevel security. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 35(5):948–958, 2005. doi:10.1109/TSMCB.2005.847749.
- 20 Christoforos N. Hadjicostis. *Estimation and Inference in Discrete Event Systems: A Model-Based Approach with Finite Automata*. Communications and Control Engineering. Springer Cham, 2020. doi:10.1007/978-3-030-30821-6.
- 21 Timon Hertli. 3-SAT faster and simpler - unique-sat bounds for PPSZ hold in general. *SIAM Journal on Computing*, 43(2):718–729, 2014. doi:10.1137/120868177.
- 22 John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to automata theory, languages, and computation*. Pearson international edition. Addison-Wesley, third edition, 2007.
- 23 Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-SAT. *Journal of Computer and System Sciences*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.
- 24 Romain Jacob, Jean-Jacques Lesage, and Jean-Marc Faure. Overview of discrete event systems opacity: Models, validation, and quantification. *Annual Reviews in Control*, 41:135–146, 2016. doi:10.1016/j.arcontrol.2016.04.015.
- 25 Feng Lin. Diagnosability of discrete event systems and its applications. *Discrete Event Dynamic Systems*, 4(2):197–212, 1994. doi:10.1007/BF01441211.
- 26 Feng Lin. Opacity of discrete event systems and its applications. *Automatica*, 47(3):496–503, 2011. doi:10.1016/j.automatica.2011.01.002.
- 27 Feng Lin and Walter Murray Wonham. On observability of discrete-event systems. *Information Sciences*, 44(3):173–198, 1988. doi:10.1016/0020-0255(88)90001-1.
- 28 Tomáš Masopust. Complexity of deciding detectability in discrete event systems. *Automatica*, 93:257–261, 2018. doi:10.1016/j.automatica.2018.03.077.
- 29 Tomáš Masopust and Xiang Yin. Complexity of detectability, opacity and A-diagnosability for modular discrete event systems. *Automatica*, 101:290–295, 2019. doi:10.1016/j.automatica.2018.12.019.
- 30 Laurent Mazaré. Decidability of opacity with non-atomic keys. In Theodosios Dimitrakos and Fabio Martinelli, editors, *Formal Aspects in Security and Trust: Second IFIP TC1 WG1.7 Workshop on Formal Aspects in Security and Trust (FAST), an event of the 18th IFIP World Computer Congress, Toulouse, France, August 22-27, 2004*, volume 173 of *IFIP*, pages 71–84. Springer, 2004. doi:10.1007/0-387-24098-5\_6.
- 31 Ramamohan Paturi, Pavel Pudlák, Michael E. Saks, and Francis Zane. An improved exponential-time algorithm for k-SAT. *Journal of the ACM*, 52(3):337–364, 2005. doi:10.1145/1066100.1066101.
- 32 Michael K. Reiter and Aviel D. Rubin. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998. doi:10.1145/290163.290168.
- 33 Andrei Sabelfeld and Andrew C. Myers. Language-based information-flow security. *IEEE Journal on Selected Areas in Communications*, 21(1):5–19, 2003. doi:10.1109/JSAC.2002.806121.

- 34 Anooshiravan Saboori. *Verification and enforcement of state-based notions of opacity in discrete event systems*. PhD thesis, University of Illinois at Urbana-Champaign, 2011.
- 35 Anooshiravan Saboori and Christoforos N. Hadjicostis. Verification of infinite-step opacity and complexity considerations. *IEEE Transactions on Automatic Control*, 57(5):1265–1269, 2012. doi:10.1109/TAC.2011.2173774.
- 36 Anooshiravan Saboori and Christoforos N. Hadjicostis. Verification of initial-state opacity in security applications of discrete event systems. *Information Sciences*, 246:115–132, 2013. doi:10.1016/j.ins.2013.05.033.
- 37 Meera Sampath, Raja Sengupta, Stéphane Lafortune, Kasim Sinnamohideen, and Demosthenis Teneketzis. Diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control*, 40(9):1555–1575, 1995. doi:10.1109/9.412626.
- 38 Steve A. Schneider and Abraham Sidiropoulos. CSP and anonymity. In Elisa Bertino, Helmut Kurth, Giancarlo Martella, and Emilio Montolivo, editors, *European Symposium on Research in Computer Security, ESORICS 1996, Rome, Italy, September 25-27, 1996*, volume 1146 of *Lecture Notes in Computer Science*, pages 198–218. Springer, 1996. doi:10.1007/3-540-61770-1\_38.
- 39 Shaolong Shu, Feng Lin, and Hao Ying. Detectability of discrete event systems. *IEEE Transactions on Automatic Control*, 52(12):2356–2359, 2007. doi:10.1109/TAC.2007.910713.
- 40 N. J. A. Sloan. The on-line encyclopedia of integer sequences (OEIS), 2023. A123121. URL: <https://oeis.org/A123121>.
- 41 N. J. A. Sloan. The on-line encyclopedia of integer sequences (OEIS), 2023. A001511. URL: <https://oeis.org/A001511>.
- 42 Andrew Wintenberg, Matthew Blischke, Stéphane Lafortune, and Necmiye Ozay. A dynamic obfuscation framework for security and utility. In *ACM/IEEE International Conference on Cyber-Physical Systems, ICCPS 2022, Milano, Italy, May 4-6, 2022*, pages 236–246. IEEE, 2022. doi:10.1109/ICCPS54341.2022.00028.
- 43 Yi-Chin Wu and Stéphane Lafortune. Comparative analysis of related notions of opacity in centralized and coordinated architectures. *Discrete Event Dynamic Systems*, 23(3):307–339, 2013. doi:10.1007/s10626-012-0145-z.
- 44 Yi-Chin Wu, Vasumathi Raman, Blake C. Rawlings, Stéphane Lafortune, and Sanjit A. Seshia. Synthesis of obfuscation policies to ensure privacy and utility. *Journal of Automated Reasoning*, 60(1):107–131, 2018. doi:10.1007/s10817-017-9420-x.
- 45 Xiang Yin and Stéphane Lafortune. A new approach for the verification of infinite-step and K-step opacity using two-way observers. *Automatica*, 80:162–171, 2017. doi:10.1016/j.automatica.2017.02.037.
- 46 Aris Zakynthinos and E. Stewart Lee. A general theory of security properties. In *IEEE Symposium on Security and Privacy, Oakland, CA, USA, May 4-7, 1997*, pages 94–102. IEEE Computer Society, 1997. doi:10.1109/SECPRI.1997.601322.

## A Proofs of Claims

▷ Claim 9. Let  $\mathcal{A}_\varphi^X$  denote  $\mathcal{A}_\varphi$  without the states  $q_s$  and  $q_{ns}$  and the corresponding transitions. For every  $w \in Z_n$ , after reading the prefix of  $w$  of length  $\ell \leq 2^n - 1$ , the configuration of  $\mathcal{A}_\varphi^X$  is  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_1^{r_1}\}$ , where  $r_n r_{n-1} \dots r_1$  represents  $\ell$  in binary.

Proof. By induction on  $\ell$ . The initial configuration of  $\mathcal{A}_\varphi^X$  is  $\{x_n^0, x_{n-1}^0, \dots, x_1^0\}$ , where  $00 \dots 0$  represents  $\ell = 0$  in binary. Assume that the configuration of  $\mathcal{A}_\varphi^X$  after reading the prefix of  $w \in Z_n$  of length  $\ell < 2^n - 1$  is  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_1^{r_1}\}$ , where  $r_n r_{n-1} \dots r_1$  represents  $\ell$  in binary. Let  $r_t$  be the rightmost zero of  $r_n r_{n-1} \dots r_1$ ; that is,  $r_n r_{n-1} \dots r_1 = r_n r_{n-1} \dots r_{t+1} 0 1 \dots 1$ . Then,  $\ell + 1$  is represented as  $r_n r_{n-1} \dots r_{t+1} 1 0 \dots 0$  in binary, and because it has  $t - 1$  trailing zeros, the  $(\ell + 1)$ st symbol of  $w$  is of the form  $\{a_t\} \times C$  by (1). It remains to show that every  $(a_t, c) \in \{a_t\} \times C$  moves  $\mathcal{A}_\varphi^X$  from the configuration  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_{t+1}^{r_{t+1}}, x_t^0, x_{t-1}^1, \dots, x_1^1\}$  to the

configuration  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_{t+1}^{r_{t+1}}, x_t^1, x_{t-1}^0, \dots, x_1^0\}$ . By the definition of  $\mathcal{A}_\varphi^X$ , the transition under  $(a_t, c)$  is undefined in states  $x_{t-1}^1, \dots, x_1^1$ , it is a self-loop in states  $x_n^{r_n}, \dots, x_{t+1}^{r_{t+1}}$ , and it moves  $\mathcal{A}_\varphi^X$  from state  $x_t^0$  to states  $x_t^1$  and  $x_{t-1}^0, \dots, x_1^0$ . Therefore, the automaton  $\mathcal{A}_\varphi^X$  moves to the configuration  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_{t+1}^{r_{t+1}}, x_t^1, x_{t-1}^0, \dots, x_1^0\}$ , where  $r_n r_{n-1} \dots r_{t+1} 10 \dots 0$  represents  $\ell + 1$  in binary.  $\triangleleft$

▷ **Claim 10.** Every configuration of  $\mathcal{A}_\varphi$  contains the secret state  $q_s$ .

*Proof.* The secret state  $q_s$  is an initial state of  $\mathcal{A}_\varphi$  and since it contains a self-loop under all symbols of  $\Sigma$ , it appears in every configuration of  $\mathcal{A}_\varphi$ .  $\triangleleft$

We now show that if there is a satisfying assignment, then the non-secret state  $q_{ns}$  is reached by  $\mathcal{A}_\varphi$ . To this end, for a variable  $x$  and  $r \in \{0, 1\}$ , we define the function

$$\text{LIT}(x^r) = \begin{cases} \neg x & \text{if } r = 0 \\ x & \text{if } r = 1 \end{cases}$$

▷ **Claim 12.** For every prefix  $w$  of a string in  $Z_n$ , if the configuration of  $\mathcal{A}_\varphi$  after reading  $w$  satisfies  $\varphi$ , then after reading any further symbol in  $\Sigma$ , the configuration of  $\mathcal{A}_\varphi$  contains  $q_{ns}$ .

*Proof.* By Claim 11, the configuration of  $\mathcal{A}_\varphi$  after reading  $w$  is  $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \dots, x_1^{r_1}\} \cup Y$ , where  $Y \in \{\{q_s\}, \{q_s, q_{ns}\}\}$ . Since the assignment  $r_n r_{n-1} \dots r_1$  satisfies  $\varphi$ , for every  $c \in C$ , there is  $i$  such that  $\text{LIT}(x_i^{r_i})$  satisfies  $c$ , and hence there are transitions from  $x_i^{r_i}$  to  $q_{ns}$  under  $(a_j, c)$  for all  $j = 1, \dots, n + 1$ . Therefore, after reading any symbol from  $\Sigma$ , the configuration of  $\mathcal{A}_\varphi$  contains  $q_{ns}$ .  $\triangleleft$

▷ **Claim 13.** If  $\varphi$  is not satisfiable, there is a string  $w_\varphi \in Z_n$  such that the configuration of  $\mathcal{A}_\varphi$  after reading  $w_\varphi$  is  $\{x_n^1, x_{n-1}^1, \dots, x_1^1\} \cup \{q_s\}$ .

*Proof.* We construct a sequence  $\varepsilon = w_0, w_1, \dots, w_{2^n-1} = w_\varphi$  of prefixes of the required string such that, for  $\ell = 0, \dots, 2^n - 1$ , the configuration of  $\mathcal{A}_\varphi$  after reading  $w_\ell$  is  $\{x_n^{r_n}, \dots, x_1^{r_1}\} \cup \{q_s\}$  and  $r_n \dots r_1$  is the representation of  $\ell$  in binary.

We proceed by induction on  $\ell$ . Since the initial configuration of  $\mathcal{A}_\varphi$  is  $\{x_n^0, x_{n-1}^0, \dots, x_1^0\} \cup \{q_s\}$ , the claim holds for  $\ell = 0$ . Now, assume that the claim holds for  $w_\ell$  with  $\ell < 2^n - 1$ , and denote the configuration of  $\mathcal{A}_\varphi$  after reading  $w_\ell$  by  $\{x_n^{r_n}, \dots, x_1^{r_1}\} \cup \{q_s\}$ . We show that there is  $(a, c) \in \Sigma$  such that  $w_{\ell+1} = w_\ell(a, c)$ . Let  $r_t$  be the rightmost 0 of  $r_n \dots r_1$  and take  $a = a_t$ . Then, for any choice of  $c$ , the string  $w_{\ell+1}$  is, by (1), a prefix of a string in  $Z_n$ , and, by Claim 9, the configuration of  $\mathcal{A}_\varphi$  after reading  $w_{\ell+1}$  is  $\{x_n^{r'_n}, \dots, x_1^{r'_1}\} \cup Y$ , where  $Y \in \{\{q_s\}, \{q_s, q_{ns}\}\}$  and  $r'_n \dots r'_1$  represents  $\ell + 1$  in binary. Since  $\varphi$  is not satisfiable, there is a clause  $c'$  that is not satisfied by  $\text{LIT}(x_i^{r_i})$  for any  $i = 1, \dots, n$ . Taking  $c = c'$  then gives the required symbol. Indeed, the transition from  $x_i^{r_i}$  under  $(a_t, c')$  is undefined for  $i = 1, \dots, t - 1$ , it is a self-loop for  $i = t + 1, \dots, n$ , and it takes  $x_t^0$  to  $\{x_t^1\} \cup \{x_j^0 \mid 1 \leq j \leq t - 1\}$ , see Figure 1; therefore,  $Y = \{q_s\}$ .  $\triangleleft$