On Verification of Strong Periodic D-Detectability for Discrete Event Systems ⋆

Jiří Balun * and **Tomáš Masopust** *,**

* *Department of Computer Science, Faculty of Science, Palacky University, 17. listopadu 12, 771 46 Olomouc, Czechia*
** *Institute of Mathematics, Czech Academy of Sciences, Brno, Czechia (e-mails: jiri.balun01@upol.cz, tomas.masopust@upol.cz)*

Abstract: Detectability of discrete event systems has been introduced as a generalization of other state-estimation properties studied in the literature, including stability or observability. It is a property whether the current and subsequent states of a systems can be determined based on observations. Since the requirement to exactly determine the current and subsequent states may be too strict in some applications, a relaxed notion of D-detectability has been introduced, distinguishing only certain pairs of states rather than all states. Four variants of D-detectability have been defined: strong (periodic) D-detectability and weak (periodic) D-detectability. The complexity of verifying weak (periodic) D-detectability follows from the results for verifying detectability, and hence is PSpace-complete. Similarly, Shu and Lin constructed a detector that can check, in polynomial time, both strong (periodic) detectability and strong D-detectability. However, the case of strong periodic D-detectability is more involved, and, to the best of our knowledge, the question whether it can be verified in polynomial time is open. We answer this question by showing that there is no algorithm verifying the strong periodic D-detectability property in polynomial time, unless every problem solvable in polynomial space can be solved in polynomial time. Hence, the algorithm based on the construction of the observer is the best possible. We further show that the strong periodic D-detectability property cannot be verified in polynomial time even for systems having only a single observable event, unless P = NP.

Keywords: Discrete event systems, finite-state automata, detectability, verification, complexity

1. INTRODUCTION

Detectability of discrete event systems (DESs) modeled by finite-state automata has been introduced by Shu et al. (2007) as a generalization of other notions studied in the literature, including stability of Ozveren and Willsky (1990) or observability of Caines et al. (1988) or of Ramadge (1986). An evidence that many practical problems can be formulated as the detectability problem for DESs has been provided by Shu and Lin (2011). In addition, Lin (2011) has shown that the detectability property is closely related to other important properties, such as observability, diagnosability, and opacity.

Detectability is a state-estimation property asking whether the current and subsequent states of a DES can be determined after a finite number of observations. Shu et al. (2007) have defined four variants of the detectability problem: strong (periodic) detectability and weak (periodic) detectability. In their work, they first studied the detectability problem for deterministic DESs, which are deterministic finite-state automata with a set of initial states rather than a single initial state. The motivation for a set of initial states results from the observation that it is often unknown which state the system is initially in. They proposed an exponential algorithm for checking the detectability property of a DES based on the computation of the observer. Shortly after, Shu and Lin (2011) extended the problem to nondeterministic DESs and designed a new algorithm that checks the strong (periodic) detectability property of nondeterministic DESs in polynomial time. Masopust (2018a) later studied the computation complexity of the problem and showed that checking the strong (periodic) detectability property is an NL-complete problem. Consequently, the problem is efficiently solvable on a parallel computer.

The complexity of verifying the weak (periodic) detectability property has been investigated only recently. Zhang (2017) has shown that the problem is PSpace-complete and that it remains PSpace-hard even for deterministic DESs with all events observable. Masopust (2018a) further strengthened these results by proving the same complexity for structurally "simplest" deadlock-free DESs that are modeled by deterministic finite-state automata without non-trivial cycles.

Since the requirement in the definition of detectability to exactly determine the current and subsequent states after a finite number of observations may be too strict in some applications, Shu and Lin (2011) relaxed the notion of detectability to a so-called *D-detectability* property. The idea behind the relaxation is to distinguish only certain pairs of states rather than all states of the system. Four

⋆ This work was partially supported by the Czech Science Foundation under grant No. GC19-06175J and by RVO 67985840.

variants of D-detectability have been defined: strong (periodic) D-detectability and weak (periodic) D-detectability.

The notion of (D-)detectability has been extended in many directions. To mention a few, Shu and Lin (2013) extended strong (D-)detectability to delayed (D-)detectability, motivated by discrete event systems with delays, and designed a polynomial-time algorithm to check strong (D-)detectability for delayed DESs. Zhang and Giua (2019) have recently improved the algorithm for checking strong delayed (D-)detectability. They further introduced several other notions of detectability, see Zhang et al. (2020) for more details. Alves and Basilio (2019) recently studied (D-)detectability for discrete event systems with multi-channel communication networks. Yin and Lafortune (2017) examined the verification of the weak and strong detectability properties for modular DESs, and showed that checking both the strong modular detectability and the weak modular detectability property is PSpace-hard. The exact complexities of these two problems have recently been shown by Masopust and Yin (2019). They are, respectively, PSpace-complete and ExpSpace-complete. We refer the reader to Hadjicostis (2020) for the latest development of state-estimation properties.

The complexity of checking whether a DES satisfies the weak (periodic) D-detectability property follows directly from the complexity of checking the weak (periodic) detectability property, and is thus PSpace-complete.

The case of strong D-detectability is similar to that of strong detectability. For strong (periodic) detectability, Shu and Lin (2011) designed a detector that can check, in polynomial time, whether a DES satisfies the strong (periodic) detectability property. They have further shown that their detector is also suitable for checking the strong D-detectability property. Consequently, the complexity of verifying whether a DES satisfies the strong D-detectability property is polynomial; see Zhang and Giua (2019) for details on the algorithmic complexity.

We further improve this result by showing that deciding whether a DES satisfies the strong D-detectability property is an NL-complete problem (Theorem 1). Since NL is the class of problems that can be efficiently parallelized, see Arora and Barak (2009) for details, we obtain that the verification of strong D-detectability can be efficiently verified on a parallel computer.

However, the verification of strong periodic D-detectability is more involved. Although the detector-based technique gives a polynomial-time algorithm to decide whether a DES satisfies strong periodic detectability, Shu and Lin (2011) present an example that this algorithm does not work for checking strong periodic D-detectability. They leave the question of the existence of a polynomial-time algorithm checking the strong periodic D-detectability property of a DES open. To the best of our knowledge, this question has not yet been answered in the literature. We answer this open question by showing that there does not exist any algorithm that would, in polynomial time, verify whether a DES satisfies strong periodic D-detectability (Theorem 3), unless P = PSpace. The question whether P = PSpace is a longstanding open problem of computer science. In other words, it asks whether every problem solvable in polynomial space can also be solved in polynomial time. It is generally believed that it is not the case. Theorem 3 shows that the strong periodic D-detectability problem is PSpace-complete. Formulated differently, the result says that the technique based on the computation of the observer is basically optimal. Since NL is a strict subclass of PSpace, the strong periodic D-detectability problem is significantly more complex than its non-periodic counterpart—strong D-detectability.

Finally, we point out that the strong periodic D-detectability property is more complex than its non-periodic counterpart even for systems having only a single observable event. More specifically, we show that strong periodic D-detectability cannot be verified in polynomial time even for DESs that have only a single observable event (Theorem 7), unless P = NP.

2. PRELIMINARIES AND DEFINITIONS

For a set A, $|A|$ denotes the cardinality of A and 2^A its power set. An alphabet Σ is a finite nonempty set of events. A string over Σ is a sequence of events of Σ. Let Σ^* denote the set of all finite strings over Σ; the empty string is denoted by ε. For a string $u \in \Sigma^*$, $|u|$ denotes its length. As usual, the notation Σ^+ stands for $\Sigma^* \setminus \{\varepsilon\}$.

A *nondeterministic finite-state automaton* (NFA) over an alphabet Σ is a structure $\mathcal{A} = (Q, \Sigma, \delta, I, F)$, 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. The *language recognized by* $\mathcal{A}$ is the set $L(\mathcal{A}) = \{w \in \Sigma^* \mid \delta(I, w) \cap F \neq \emptyset\}$. Equivalently, the transition function δ is a relation $\delta \subseteq Q \times \Sigma \times Q$, where, for instance, $\delta(q, a) = \{s, t\}$ denotes the two transitions (q, a, s) and (q, a, t).

The NFA $\mathcal{A}$ is *deterministic* (DFA) if it has a unique initial state, i.e., $|I| = 1$, and no nondeterministic transitions, i.e., $|\delta(q, a)| \leq 1$ for every $q \in Q$ and $a \in \Sigma$. The DFA $\mathcal{A}$ is *total* if in every state, a transition under every event is defined, i.e., $|\delta(q, a)| = 1$ for every $q \in Q$ and $a \in \Sigma$.

A *discrete event system* (DES) is an NFA G with all states marked. Hence we simply write $G = (Q, \Sigma, \delta, I)$ leaving out the set of marked states. Additionally, the alphabet Σ is partitioned into the set Σ_o of *observable events* and the set $\Sigma_{uo} = \Sigma \setminus \Sigma_o$ of *unobservable events*.

State-estimation properties are based on the observation of events. The observation is described by the projection $P\colon \Sigma^* \to \Sigma_o^*$. The projection $P\colon \Sigma^* \to \Sigma_o^*$ is a morphism defined by $P(a) = \varepsilon$ for $a \in \Sigma \setminus \Sigma_o$, and $P(a) = a$ for $a \in \Sigma_o$. The action of P on a string $w = a_1 a_2 \cdots a_n$, where $a_i \in \Sigma$ for $1 \leq i \leq n$, is to erase all events from w that do not belong to Σ_o; in particular, $P(a_1 a_2 \cdots a_n) = P(a_1) P(a_2) \cdots P(a_n)$. The definition can readily be extended to infinite strings and languages.

Shu and Lin (2011) make the following two reasonable assumptions on the DES $G = (Q, \Sigma, \delta, I)$ that we adopt:

(1) G is *deadlock free* – it means that for every state of the system, at least one event can occur; formally, for every $q \in Q$, there exists $\sigma \in \Sigma$ such that $\delta(q, \sigma) \neq \emptyset$.

(2) No loop in G consists solely of unobservable events – for every $q \in Q$ and every $w \in \Sigma_{uo}^{+}$, $q \notin \delta(q, w)$.

We point out that to verify whether a system has these two properties is very easy. The violation of any of these two properties in a system is often considered as a modelling error. Moreover, omitting the conditions does not change our results.

The set of infinite sequences of events (or trajectories) generated by the DES G is denoted by $L^{\omega}(G)$. Given a set $Q' \subseteq Q$, the set of all possible states after observing a string $t \in \Sigma_o^*$ is denoted by

$$R(Q', t) = \bigcup_{w \in \Sigma^*, P(w) = t} \delta(Q', w) \,.$$

For $w \in L^{\omega}(G)$, we denote the set of its prefixes by $Pr(w)$.

In this paper, we study the complexity questions. Therefore, we briefly recall the basics of complexity theory needed to understand the results. A *decision problem* is a yes-no question. A decision problem is *decidable* if there exists an algorithm that can solve the problem. Complexity theory classifies decidable problems to classes according to the time or space an algorithm needs to solve the problem. The complexity classes we consider in this paper are NL, P, NP, and PSpace. They denote the classes of problems solvable by a nondeterministic logarithmic-space, deterministic polynomial-time, nondeterministic polynomial-time, and deterministic polynomial-space algorithm, respectively. The hierarchy of these classes is NL $\subseteq$ P $\subseteq$ NP $\subseteq$ PSpace. Which of the inclusions are strict is a longstanding open problem in computer science. The widely accepted conjecture is that all inclusions are strict. However, so far only the inclusion NL $\subseteq$ PSpace is known to be strict. A decision problem is NL-complete (resp. NP-complete, PSpace-complete) if it belongs to NL (resp. NP, PSpace) and every problem from NL (resp. NP, PSpace) can be reduced to it by a deterministic logarithmic-space (resp. polynomial-time) algorithm.

3. THE D-DETECTABILITY PROBLEMS

Shu and Lin (2011) defined D-detectability as a generalization of detectability by making the states that need to be distinguished explicit.

Let $G = (Q, \Sigma, \delta, I)$ be a DES, and let $T_{spec} \subseteq Q \times Q$ be a symmetric relation on the set of states of G. The relation T_{spec} specifies pairs of states that must be distinguished, and is therefore called a specification. The idea behind the definition of D-detectability is to ensure that the pairs of states from T_{spec} are distinguished after a finite number of observations.

We now recall the four variants of D-detectability.

A discrete event system $G = (Q, \Sigma, \delta, I)$ is *strongly D-detectable* with respect to the projection $P \colon \Sigma^* \to \Sigma_o^*$ and the specification T_{spec} if, for all trajectories of the system, the pairs of states of T_{spec} can be distinguished in every step of the system after a finite number of observations. This is formally defined as follows:

$$(\exists n \in \mathbb{N})(\forall s \in L^{\omega}(G))(\forall t \in Pr(s))\, |P(t)| > n \Rightarrow (R(Q_0, P(t)) \times R(Q_0, P(t))) \cap T_{spec} = \emptyset \,.$$

A discrete event system $G = (Q, \Sigma, \delta, I)$ is *weakly D-detectable* with respect to the projection $P \colon \Sigma^* \to \Sigma_o^*$ and the specification T_{spec} if, for some trajectories of the system, the pairs of states of T_{spec} can be distinguished in every step of the system after a finite number of observations. This is formally defined as follows:

$$(\exists n \in \mathbb{N})(\exists s \in L^{\omega}(G))(\forall t \in Pr(s))\, |P(t)| > n \Rightarrow (R(Q_0, P(t)) \times R(Q_0, P(t))) \cap T_{spec} = \emptyset \,.$$

A discrete event system $G = (Q, \Sigma, \delta, I)$ is *strongly periodically D-detectable* with respect to the projection $P \colon \Sigma^* \to \Sigma_o^*$ and the specification T_{spec} if the pairs of states of T_{spec} can be periodically distinguished for all trajectories of the system. Formally,

$$(\exists n \in \mathbb{N})(\forall s \in L^{\omega}(G))(\forall t \in Pr(s))(\exists t' \in \Sigma^*)\; tt' \in Pr(s) \wedge |P(t')| < n \wedge (R(Q_0, P(tt')) \times R(Q_0, P(tt'))) \cap T_{spec} = \emptyset \,.$$

A discrete event system $G = (Q, \Sigma, \delta, I)$ is *weakly periodically D-detectable* with respect to the projection $P \colon \Sigma^* \to \Sigma_o^*$ and the specification T_{spec} if the pairs of states of T_{spec} can be periodically distinguished for some trajectories of the system. Formally,

$$(\exists n \in \mathbb{N})(\exists s \in L^{\omega}(G))(\forall t \in Pr(s))(\exists t' \in \Sigma^*)\; tt' \in Pr(s) \wedge |P(t')| < n \wedge (R(Q_0, P(tt')) \times R(Q_0, P(tt'))) \cap T_{spec} = \emptyset \,.$$

4. RESULTS

In this section, for all the variants of the D-detectability problem, we discuss the complexity of deciding whether a DES satisfies the given D-detectability property.

As already pointed out in the introduction, the complexity of checking whether a DES satisfies weak (periodic) D-detectability follows directly from the complexity of checking weak (periodic) detectability. Indeed, a polynomial space is sufficient for an algorithm based on the inspection of states in the observer and works for all the D-detectability variants. Therefore, the problem of deciding whether a DES satisfies weak (periodic) D-detectability belongs to PSpace. On the other hand, detectability is a special case of D-detectability for $T_{spec} = Q \times Q \setminus \{(q, q) \mid q \in Q\}$. Therefore, deciding weak (periodic) D-detectability is at least as hard as deciding weak (periodic) detectability. Since the later problem is PSpace-hard, so is the former.

4.1 Verification of Strong D-Detectability

Shu and Lin (2011) designed an algorithm that verifies strong (periodic) detectability in polynomial time. Their algorithm is based on the construction of a finite-state automaton called a *detector*. Intuitively, given a DES G, their detector G_{det} is constructed from G so that

- the set of initial states of G_{det} is the set of all states of G reachable from the initial states of G under strings consisting only of unobservable events,
- all the other states of G_{det} are one- or two-element subsets of the set of states of G, and
- the transition relation of G_{det} is constructed in the similar way as that of the observer, but if the reached

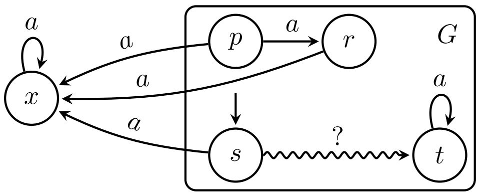

Fig. 1. The DES $\mathcal{A}$ constructed from G in the NL-hardness proof of Theorem 1

state X in the observer consists of more than two states, then the detector G_{det} has several transitions each leading to a two-element subset of X, see Shu and Lin (2011) for details.

Since the states of the detector are one- or two-element subsets, their number is polynomial. Shu and Lin (2011) showed that a DES G satisfies strong (periodic) detectability if and only if any state reachable from any loop in G_{det} consists solely (periodically) of distinguishable states. They further proved that their algorithm, respectively the detector, works for checking whether a DES satisfies strong D-detectability. This in particular implies that the complexity of verifying whether a DES satisfies strong D-detectability is polynomial. Zhang and Giua (2019) recently improved the algorithmic complexity of this problem.

We now discuss the computational complexity of deciding strong D-detectability and show that it is an NL-complete problem. Consequently, since NL is the class of problems that can be efficiently parallelized, see Arora and Barak (2009) for details, our result shows that the question whether a DES satisfies strong D-detectability can be efficiently verified on a parallel computer.

Theorem 1. Deciding whether a DES satisfies the strong D-detectability property is an NL-complete problem.

Proof. We prove the membership of the problem in NL by showing that checking that the condition does not hold is in NL. This also proves that checking whether the condition is satisfied is in NL, since the class NL is closed under complement, see Immerman (1988) or Szelepcsényi (1988) for details.

Thus, to check that the property is not satisfied, the NL algorithm works so that it guesses two states of G_{det}, say x and y, where y contains indistinguishable states, and verifies that (i) y is reachable from x, (ii) x is reachable from the initial state of G_{det}, and (iii) x is in a cycle, i.e., x is reachable from x by a path having at least one transition. Notice that the algorithm does not construct the detector G_{det}. It only stores a constant number of states of G_{det} and computes the required transitions of G_{det} on demand. Therefore, the algorithm does not need more that a logarithmic space. For more details how to check reachability in NL, we refer the reader to Masopust (2018b).

To show NL-hardness, we reduce the *DAG non-reachability problem*, see Cho and Huynh (1991) for details: Given a directed acyclic graph $G = (V, E)$ and two nodes $s, t \in V$, it asks whether t is not reachable from s.

From G, we construct a DES $\mathcal{A} = (V \cup \{x\}, \{a\}, \delta, s)$, where $x \notin V$ is a new state and a is an observable event.

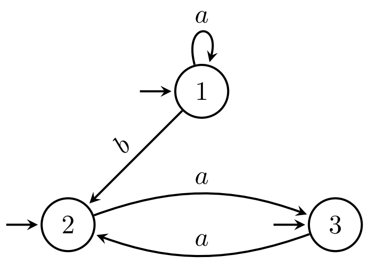

Fig. 2. The DES G from Example 2

For every $(p, r) \in E$, we add the transition (p, a, r) to δ, and for every $p \in V \setminus \{t\}$, we add the transition (p, a, x) to δ. Moreover, we add the self-loop transitions (x, a, x) and (t, a, t) to δ. The construction is depicted in Fig. 1. Notice that $\mathcal{A}$ is deadlock-free and has no unobservable events.

Finally, we let the symmetric specification T_{spec} be defined as $T_{spec} = \{(t, x), (x, t)\}$. We now show that t is not reachable from s in the graph G if and only if the DES $\mathcal{A}$ is strongly D-detectable.

If node t is not reachable from node s in G, then, for every $k \geq |V|$, $\delta(s, a^k) = \{x\}$. Therefore, $\mathcal{A}$ satisfies the strong D-detectability property.

If node t is reachable from node s in G, then, for every $k \geq |V|$, $\delta(s, a^k) = \{t, x\}$. Therefore, $\mathcal{A}$ does not satisfy strong D-detectability, because $(t, x) \in T_{spec}$. □

4.2 Verification of Strong Periodic D-Detectability

Although the detector-based technique leads to a polynomial-time algorithm to decide whether a DES satisfies the strong periodic detectability property, Shu and Lin (2011) have shown that this algorithm does not work for checking strong periodic D-detectability.

To give the reader an idea of the detector-based polynomial-time algorithm and of the problem why it does not work for checking strong periodic D-detectability, we slightly elaborate the example of Shu and Lin (2011).

Example 2. Let $G = (\{1, 2, 3\}, \{a, b\}, \delta, \{1, 2, 3\})$ be the DES depicted in Fig. 2, where both events are observable. Let the specification $T_{spec} = \{(1, 3), (3, 1)\}$. The detector G_{det} is depicted in Fig. 3.

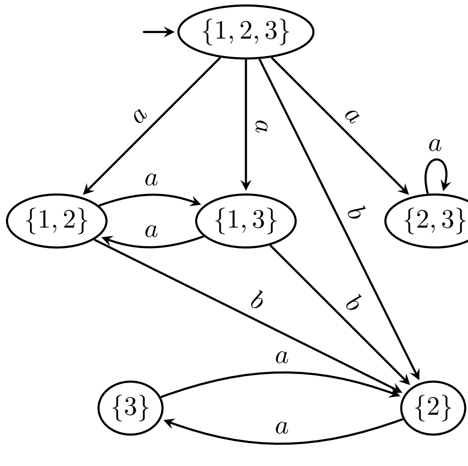

Fig. 3. The detector G_{det} constructed from the DES G from Example 2

In the detector G_{det}, we can immediately see that G does not satisfy the strong D-detectability property, since there is an infinite path that goes periodically through the state $\{1, 3\}$, violating thus strong D-detectability.

From the same infinite path going periodically through the states $\{1, 2\}$ and $\{1, 3\}$, the reader could get an impression that G satisfies the strong periodic D-detectability property. However, this is not the case as can be easily seen from the observer depicted in Fig. 4. There, there is an infinite path a^* in state $\{1, 2, 3\}$, which violates the strong periodic D-detectability property. ⋄

Shu and Lin (2011) have left the question whether there exists a polynomial-time algorithm checking the strong periodic D-detectability property of a DES open. And, to the best of our knowledge, this question has not yet been answered in the literature. We answer this question in the sequel.

We distinguish two cases based on the number of observable events in the system:

(i) The general case where the system has two or more observable events;
(ii) A special case where the system has only a single observable event.

The case of two or more observable events

As already discussed above, the problem whether a DES satisfies strong or weak (periodic) D-detectability belongs to PSpace. In this section, we show that the problem of deciding strong periodic D-detectability is PSpace-hard, and hence PSpace-complete. Consequently, there is no algorithm solving this problem in polynomial time, unless P = PSpace.

Theorem 3. Deciding whether a DES G with at least two observable events satisfies the strong periodic D-detectability property is a PSpace-complete problem.

Proof. To show PSpace-hardness, we reduce the intersection emptiness problem: Given a sequence $\mathcal{A}_1, \ldots, \mathcal{A}_n$ of total deterministic finite-state automata over a common alphabet, is the language $\cap_{i=1}^{n} L(\mathcal{A}_i) = \emptyset$? This problem is PSpace-complete, see Garey and Johnson (1990) for details.

From $\mathcal{A}_1, \ldots, \mathcal{A}_n$, we construct a DES G that is strongly periodically D-detectable if and only if the intersection of the languages of $\mathcal{A}_1, \ldots, \mathcal{A}_n$ is empty. The main idea of our proof is to construct G as a nondeterministic union of the automata $\mathcal{A}_1, \ldots, \mathcal{A}_n$, and to add n new states forming a cycle such that these states are reachable at the same time

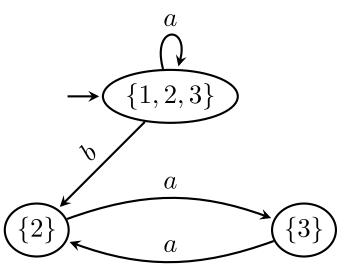

Fig. 4. The observer of the DES G from Example 2

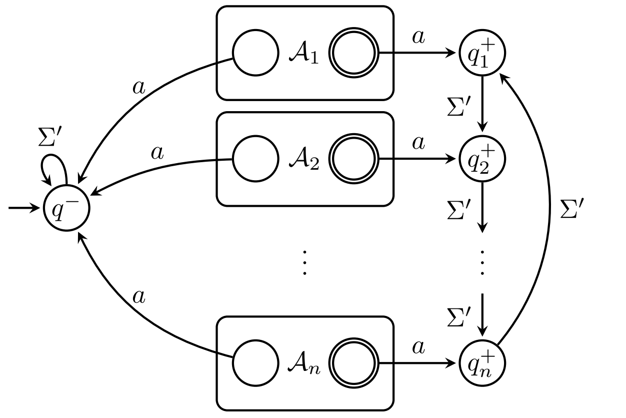

Fig. 5. Construction of the DES G from the PSpace-hardness proof of Theorem 3

if and only if the intersection is nonempty. Then, if the intersection is empty, only a strict subset of these states can be reached at the same time. After reaching these states, the computation remains only in the cycle formed by these states. During any further computation, the states are periodically rotated, which allows us to make one of them periodically indistinguishable from another state, and one periodically distinguishable. We now provide more details and two illustrative examples.

Let $\mathcal{A}_1, \ldots, \mathcal{A}_n$ be total DFAs over a common alphabet Σ, and let $\mathcal{A}_i = (Q_i, \Sigma, \delta_i, q_{0,i}, F_i)$. Without loss of generality, we may assume that the states of the DFAs are pairwise disjoint. We construct a DES G as a nondeterministic union of the automata $\mathcal{A}_i$, i.e., G contains all states and transitions of every $\mathcal{A}_i$, and we add $n + 1$ new states q^-, $q_1^+, \ldots, q_n^+$ and several new transitions under a new event $a \notin \Sigma$ as depicted in Fig. 5. Namely, for $i = 1, \ldots, n$, we add the transition (q, a, q^-) for every non-marked state $q \in Q_i \setminus F_i$, and the transition (q, a, q_i^+) for every marked state $q \in F_i$. Additionally, we add the self-loop (q^-, σ, q^-) for every $\sigma \in \Sigma' = \Sigma \cup \{a\}$. Finally, we create a cycle on the states $Q^+ = \{q_1^+, \ldots, q_n^+\}$ by adding, for every $\sigma \in \Sigma'$, the transitions $(q_i^+, \sigma, q_{i+1}^+)$, for $1 \le 1 < n$, and the transitions (q_n^+, σ, q_1^+). The set of initial states of G is the set $I = \{q^-, q_{0,1}, \ldots, q_{0,n}\}$ of the initial states of the automata $\mathcal{A}_i$ plus the newly added state q^-. The alphabet of G is $\Sigma' = \Sigma \cup \{a\}$, all events of which are observable.

We define the specification $T_{spec} = \{(q^-, q_1^+), (q_1^+, q^-)\}$, and show that G is strongly periodically D-detectable if and only if the intersection $\cap_{i=1}^{n} L(A_i)$ is empty.

Assume first that the intersection is empty. A trajectory that does not contain event a cannot violate strong periodic D-detectability, because G can never enter state q_1^+ from the specification. Thus, let $s \in L^\omega(G)$ be an arbitrary trajectory containing a. Then, after the first occurrence of event a, the observer of G is in a set of states consisting of q^- and a strict subset of Q^+; indeed, G cannot transit to all states of Q^+ at the same time, since, for every $w \in \Sigma^*$, there exists $i \in \{1, \ldots, n\}$ such that $w \notin L(A_i)$. Let $p_i \in Q^+ = \{q_1^+, \ldots, q_n^+\}$ denote the state of Q^+, with the minimal index, in which G cannot be at the ith step after reading the first occurrence of a. By construction, the cycle on Q^+ ensures that p_i periodically alternates among q_1^+

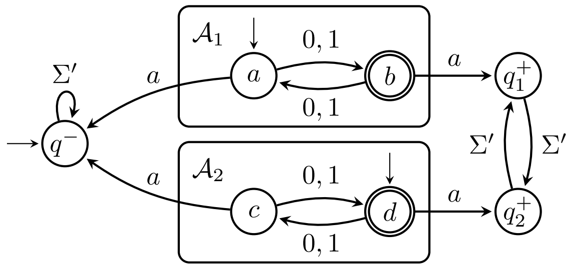

Fig. 6. The DES G_1 with $\Sigma' = \{0, 1, a\}$

and other states of Q^+ when generating events. Therefore, in the infinite sequence $p_0, p_1, \ldots$, there are infinitely many j such that $p_j = q_1^+$, and hence q^- and q_1^+ are periodically distinguished, which shows that G is strongly periodically D-detectable.

On the other hand, assume that the intersection is nonempty, and let $w \in \cap_{i=1}^{n} L(\mathcal{A}_i)$. Then, after generating the string wa, the observer of G reaches the state $\{q^-\} \cup Q^+$. Now, every transition keeps G in all states of $\{q^-\} \cup Q^+$, and hence it results in a self-loop in the observer of G. However, this self-loop violates the strong periodic D-detectability property, because it contains both states q^- and q_1^+. Therefore, any trajectory $s \in L^\omega(G)$ such that wa is its prefix leads to a set of states where the states of T_{spec} can never be distinguished, and hence G is not strongly periodically D-detectable. □

To illustrate our construction, we provide two examples.

Example 4. Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two total DFAs over the alphabet $\{0, 1\}$ depicted in Fig. 6. Then, $L(\mathcal{A}_1)$ consists of strings of odd length, and $L(\mathcal{A}_2)$ of strings of even length. The DES G_1 resulting from our construction is depicted in Fig. 6. Since $L(\mathcal{A}_1) \cap L(\mathcal{A}_2) = \emptyset$, G_1 satisfies strong periodic D-detectability. This is evident from the observer of G_1 depicted in Fig. 7, because it does not contain any cycle where q^- and q_1^+ cannot be periodically distinguished.

Example 5. For our second example, let $\mathcal{A}_1$ be as in Example 4, and let $\mathcal{A}_2$ be the total DFA such that $L(\mathcal{A}_2)$ consists of strings starting with event 0, see Fig. 8. The DES G_2 constructed according to the proof of Theorem 3 is depicted in Fig. 8. In this case, $L(\mathcal{A}_1) \cap L(\mathcal{A}_2) \neq \emptyset$, and therefore G_2 does not satisfy strong periodic D-detectability; Fig. 9 shows the observer of G_2, which contains a self-loop in state $\{q^-, q_1^+, q_2^+\}$, and hence q^- and q_1^+ cannot be periodically distinguished on this trajectory.

Remark 6. We point out that the events in the proof of Theorem 3 can be encoded in binary, and hence the result holds even if the alphabet of the DES is binary. We leave the details to the extended version.

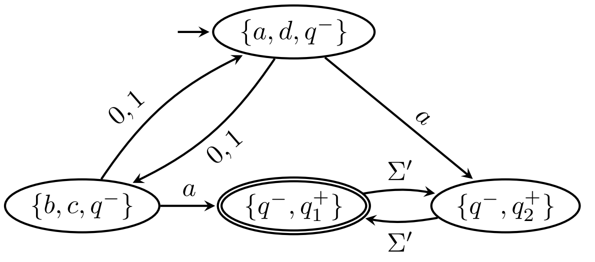

Fig. 7. The observer of the DES G_1; states marked by a double circle contain indistinguishable states of G_1

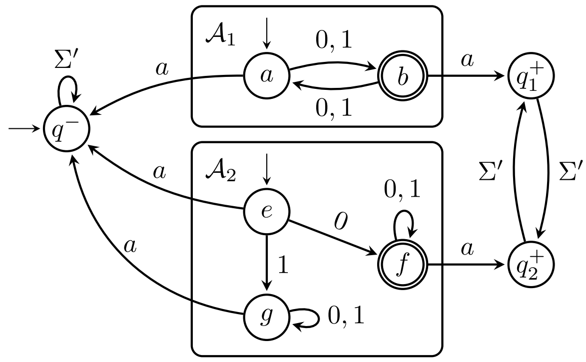

Fig. 8. The DES G_2 with $\Sigma' = \{0, 1, a\}$

The case of a single observable event

In the previous subsection, we have shown that deciding strong periodic D-detectability is PSPACE-complete if there are at least two observable events. We now show that the problem is still more difficult than its non-periodic counterpart even if the DES under consideration has a single observable event.

Theorem 7. Deciding whether a DES having only a single observable event satisfies the strong periodic D-detectability property is an NP-complete problem.

Proof sketch. A proof that the problem can be solved in nondeterministic polynomial time uses a technique based on the fast matrix multiplication that is similar to that used in Masopust (2018a). For the space reasons, we omit this proof.

We now sketch the NP-hardness proof that implies the claimed non-existence of a polynomial-time algorithm. The main idea of the proof follows the construction of Stockmeyer and Meyer (1973) encoding a boolean formula in 3CNF in the form of a unary nondeterministic finite-state automaton.

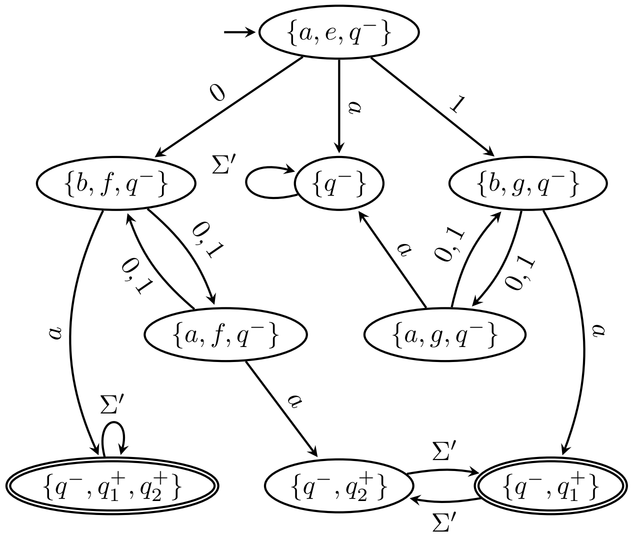

Fig. 9. The observer of the DES G_2; states marked by a double circle contain indistinguishable states of G_2

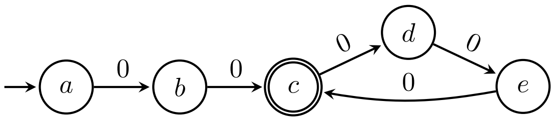

Fig. 10. Automaton $\mathcal{A}_{1,0}$

A boolean formula is built from propositional variables, operators conjunction, disjunction, and negation, and parentheses. A formula is satisfiable if there is an assignment of 1 (`true`) and 0 (`false`) to its variables making it `true`. 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; for example, $\varphi = (x \vee y \vee z) \wedge (\neg x \vee y \vee z)$ is a formula in CNF with two clauses $x \vee y \vee z$ and $\neg x \vee y \vee z$. If every clause has at most three literals, the formula is in 3CNF. Given a formula in 3CNF, the 3CNF satisfiability problem asks whether the formula is satisfiable. The formula φ is satisfiable for $(x, y, z) = (0, 1, 0)$. The 3CNF satisfiability problem is NP-complete.

Let φ be a formula in 3CNF with n variables and m clauses, and let C_k be the set of literals in the kth clause, for $k = 1, \dots, m$. The assignment to the variables can be represented as a binary vector of length n. Let $p_1, \dots, p_n$ denote the first n prime numbers. For a natural number z congruent with 0 or 1 modulo p_i, for every $i = 1, \dots, n$, we say that z satisfies φ if the assignment

$$(z \bmod p_1, z \bmod p_2, \dots, z \bmod p_n) \in \{0,1\}^n$$

satisfies φ. Let $\mathcal{A}_0$ be the nondeterministic finite-state automaton recognizing the language

$$\bigcup_{i=1}^{n} \bigcup_{j=2}^{p_i-1} 0^j \cdot (0^{p_i})^*$$

of all natural numbers that do not encode an assignment to the variables. Furthermore, for each conjunct C_k, we construct an NFA $\mathcal{A}_k$ such that if $0^z \in L(\mathcal{A}_k)$ and z is an assignment, then z does not assign value 1 (`true`) to any literal in C_k.

Having constructed all the automata $\mathcal{A}_0, \mathcal{A}_1, \dots, \mathcal{A}_k$, we take all the $\mathcal{A}_i$ altogether as a single NFA $\mathcal{A}$. We claim that the construction can be done in polynomial time, and that φ is satisfiable if and only if the NFA $\mathcal{A}$ satisfies the strong periodic D-detectability property with respect to the specification T_{spec} that is a symmetric closure of the relation consisting of pairs of states, where each pair consists of states of two different automata $\mathcal{A}_i$ and $\mathcal{A}_j$, and at least one state of every pair is a marked state of one of the automata.

Rather than to formally proof our claim, we provide two examples illustrating the construction. Full formal proofs will be provided in the full version.

Let $\varphi_1 = (x \vee y) \wedge (\neg x \vee y)$ and $\varphi_2 = x \wedge \neg x$. Obviously, φ_1 is satisfiable and φ_2 is not. For both formulae, we now construct the unary automata $\mathcal{A}_1$ and $\mathcal{A}_2$, respectively, and show that $\mathcal{A}_1$ satisfies the strong periodic D-detectability property while $\mathcal{A}_2$ does not.

Thus, consider the formula $\varphi_1 = (x \vee y) \wedge (\neg x \vee y)$. The formula has two variables, and therefore we set $p_1 = 2$ and $p_2 = 3$, the first two prime numbers. Then, the language of the automaton $\mathcal{A}_{1,0}$ is $0^2(0^3)^*$, and the automaton is depicted in Fig. 10. Since φ_1 consists of two clauses, we construct two automata: $\mathcal{A}_{1,1}$ recognizing the language $(0^6)^*$, and $\mathcal{A}_{1,2}$ recognizing the language $0^3(0^6)^*$; the reader can verify that if $0^z \in L(\mathcal{A}_{1,1})$ and z is an assignment, then z assigns `true` neither to x nor to y, and if $0^z \in L(\mathcal{A}_{1,2})$ and z is an assignment, then z assigns `true` to x (that is, it assigns `false` to the literal $\neg x$) and `false` to y. The automata are depicted in Figs. 11 and 12, respectively. The specification T_{spec} is constructed as the symmetric closure of the following relation

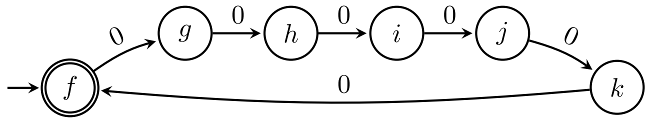

Fig. 11. Automaton $\mathcal{A}_{1,1}$

$$\begin{aligned} &\{c\} \times \{f,g,h,i,j,k,l,m,n,o,p,q,r,s,t\} \\ &\cup \{a,b,c,d,e,l,m,n,o,p,q,r,s,t\} \times \{f\} \\ &\cup \{a,b,c,d,e,f,g,h,i,j,k\} \times \{o\}\,. \end{aligned}$$

Let $\mathcal{A}_1$ be the NFA consisting of the automata $\mathcal{A}_{1,0}$, $\mathcal{A}_{1,1}$, and $\mathcal{A}_{1,2}$. The observer of $\mathcal{A}_1$ is depicted in Fig. 13. The reader can see that the observer contains a cycle where the state $\{e, j, p\}$ appears periodically. Since this state does not contain any pair from the specification T_{spec}, the NFA $\mathcal{A}_1$ satisfies strong periodic D-detectability as claimed.

Similarly, we construct the automata for the formula $\varphi_2 = x \wedge \neg x$. The formula has one variable, and therefore we set $p_1 = 2$. Then, $L(\mathcal{A}_0) = \emptyset$, and as such consists of a single non-marked state; therefore, we can ignore it in the following construction. Since φ_2 consists of two clauses, we construct two automata $\mathcal{A}_{2,1}$ recognizing the language $(0^2)^*$, and $\mathcal{A}_{2,2}$ recognizing the language $0(0^2)^*$; the reader can again verify that if $0^z \in L(\mathcal{A}_{2,1})$ and z is an assignment, then z does not assign `true` to x, and that if $0^z \in L(\mathcal{A}_{2,2})$ and z is an assignment, then z assigns `true` to x. The automata are depicted in Figs. 14 and 15, respectively. The specification T_{spec} is then the symmetric closure of the relation $\{(a,c),(a,d),(a,e),(b,d)\}$. Let $\mathcal{A}_2$ be the NFA consisting of both automata $\mathcal{A}_{2,1}$ and $\mathcal{A}_{2,2}$. The observer of $\mathcal{A}_2$ is depicted in Fig. 16. The reader can see that the state set of the observer consists of states that are indistinguishable by the specification T_{spec}, and hence $\mathcal{A}_2$ does not satisfy strong periodic D-detectability. $\square$

5. CONCLUSIONS

In this paper, we have answered the open problem concerning the complexity of deciding whether a DES satisfies the strong periodic D-detectability property. Since our study gives negative results and mainly focuses on the worst-case complexity of the problem, in the future work, we provide more details about some special cases for which the complexity of the problem is tractable.

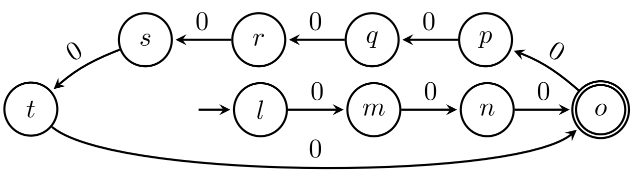

Fig. 12. Automaton $\mathcal{A}_{1,2}$

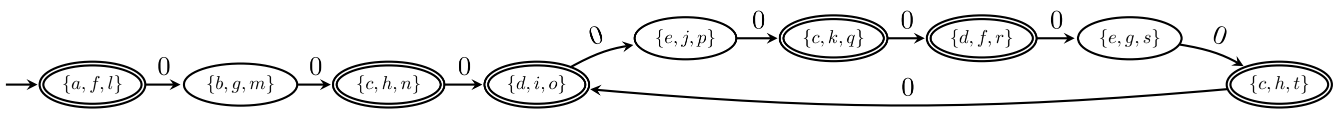

Fig. 13. The observer of the NFA $\mathcal{A}_1$; states marked by a double circle contain indistinguishable states of $\mathcal{A}_1$

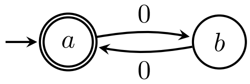

Fig. 14. Automaton $\mathcal{A}_{2,1}$

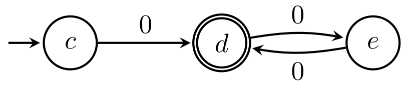

Fig. 15. Automaton $\mathcal{A}_{2,2}$

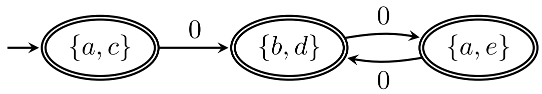

Fig. 16. The observer of the DES $\mathcal{A}_2$; states marked by a double circle contain indistinguishable states of $\mathcal{A}_2$

REFERENCES

Alves, M.V.S. and Basilio, J.C. (2019). State estimation and detectability of networked discrete event systems with multi-channel communication networks. In *American Control Conference (ACC)*, 5602–5607. IEEE. doi: 10.23919/ACC.2019.8814302.

Arora, S. and Barak, B. (2009). *Computational Complexity – A Modern Approach*. Cambridge University Press.

Caines, P.E., Greiner, R., and Wang, S. (1988). Dynamical logic observers for finite automata. In *Conference on Decision and Control (CDC)*, 226–233. doi: 10.1109/CDC.1988.194300.

Cho, S. and Huynh, D.T. (1991). Finite-automaton aperiodicity is PSPACE-complete. *Theoretical Computer Science*, 88(1), 99–116. doi:10.1016/0304-3975(91)90075-D.

Garey, M.R. and Johnson, D.S. (1990). *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA.

Hadjicostis, C.N. (2020). *Estimation and Inference in Discrete Event Systems*. Springer. doi:10.1007/978-3-030-30821-6.

Immerman, N. (1988). Nondeterministic space is closed under complementation. *SIAM Journal on Computing*, 17, 935–938. doi:10.1137/0217058.

Lin, F. (2011). Opacity of discrete event systems and its applications. *Automatica*, 47(3), 496–503. doi: 10.1016/j.automatica.2011.01.002.

Masopust, T. (2018a). Complexity of deciding detectability in discrete event systems. *Automatica*, 93, 257–261. doi:10.1016/j.automatica.2018.03.077.

Masopust, T. (2018b). Complexity of verifying nonblockingness in modular supervisory control. *IEEE Transactions on Automatic Control*, 63(2), 602–607. doi: 10.1109/TAC.2017.2726582.

Masopust, T. and Yin, X. (2019). Complexity of detectability, opacity and A-diagnosability for modular discrete event systems. *Automatica*, 101, 290–295. doi: 10.1016/j.automatica.2018.12.019.

Ozveren, C.M. and Willsky, A.S. (1990). Observability of discrete event dynamic systems. *IEEE Transactions on Automatic Control*, 35(7), 797–806. doi: 10.1109/9.57018.

Ramadge, P.J. (1986). Observability of discrete event systems. In *Conference on Decision and Control (CDC)*, 1108–1112. doi:10.1109/CDC.1986.267551.

Shu, S. and Lin, F. (2011). Generalized detectability for discrete event systems. *Systems & Control Letters*, 60(5), 310–317. doi:10.1016/j.sysconle.2011.02.001.

Shu, S. and Lin, F. (2013). Delayed detectability of discrete event systems. *IEEE Transactions on Automatic Control*, 58(4), 862–875. doi:10.1109/TAC.2012.2224255.

Shu, S., Lin, F., and Ying, H. (2007). Detectability of discrete event systems. *IEEE Transactions on Automatic Control*, 52(12), 2356–2359. doi: 10.1109/TAC.2007.910713.

Stockmeyer, L.J. and Meyer, A.R. (1973). Word problems requiring exponential time: Preliminary report. In *ACM Symposium on the Theory of Computing (STOC)*, 1–9. ACM. doi:10.1145/800125.804029.

Szelepcsényi, R. (1988). The method of forced enumeration for nondeterministic automata. *Acta Informatica*, 26, 279–284. doi:10.1007/BF00299636.

Yin, X. and Lafortune, S. (2017). Verification complexity of a class of observational properties for modular discrete events systems. *Automatica*, 83, 199–205. doi: 10.1016/j.automatica.2017.06.013.

Zhang, K. (2017). The problem of determining the weak (periodic) detectability of discrete event systems is PSPACE-complete. *Automatica*, 81, 217–220. doi: 10.1016/j.automatica.2017.03.023.

Zhang, K. and Giua, A. (2019). Revisiting delayed strong detectability of discrete-event systems. Preprint at arXiv:1910.13768.

Zhang, K., Zhang, L., and Xie, L. (2020). *Discrete-Time and Discrete-Space Dynamical Systems*. Springer. doi: 10.1007/978-3-030-25972-3.