Revisiting delayed strong detectability of discrete-event systems

Kuize Zhang · Alessandro Giua

Received: date / Accepted: date

Abstract Among notions of detectability for a discrete-event system (DES), strong detectability implies that after a finite number of observations to every output/label sequence generated by the DES, the current state can be uniquely determined. This notion is strong so that by using it the current state can be easily determined. In order to keep the advantage of strong detectability and weaken its disadvantage, we can additionally take some "subsequent outputs" into account in order to determine the current state. Such a modified observation will make some DES that is not strongly detectable become "strongly detectable in a weaker sense", which we call "$K$-delayed strong detectability" if we observe at least $K$ outputs after the time at which the state need to be determined. In this paper, we study $K$-delayed strong detectability for DESs modeled by finite-state automata (FSAs), and give a polynomial-time verification algorithm by using a novel concurrent-composition method. Note that the algorithm applies to all FSAs. Also by the method, an upper bound for $K$ has been found, and we also obtain polynomial-time verification algorithms for $(k_1, k_2)$-detectability and $(k_1, k_2)$-D-detectability of FSAs firstly studied by [Shu and Lin, 2013]. Our algorithms run in quartic polynomial time and apply to all FSAs, are more effective than the sextic polynomial-time verification algorithms given by [Shu and Lin 2013] based on the usual assumptions of deadlock-freeness and having no unobservable reachable cycle. Finally, we obtain polynomial-time synthesis algorithms for enfor-
ing delayed strong detectability, which are more effective than the exponential-time synthesis algorithms in the supervisory control framework in the literature.

**Keywords** Discrete-event system · Finite-state automaton · Delayed strong detectability · Verification · Synthesis

1 Introduction

Detectability is a basic property of dynamic systems: when it holds an observer can use the current and past values of the observed output/label sequence produced by a system to reconstruct its current state \([15,12,13,19,21,6,5]\). This property plays a fundamental role in many related control problems such as observer design and controller synthesis. On the other hand, detectability is strongly related to many cyber-security properties. For example, the property of opacity, which has been originally proposed to describe information flow security in computer science in the early 2000s \([7]\) can be seen as the absence of detectability. As another example, the detection and identification of cyber-attacks is just a particular application of detectability analysis \([8]\).

For discrete-event systems (DESs) modeled by finite-state automata\(^1\) (FSAs), the detectability problem has been widely studied \([15,12,21,6,13]\) in the context of \(\omega\)-languages, i.e., taking into account all output sequences of infinite length generated by a DES. These results are usually based on two assumptions that a system is deadlock-free and that it cannot generate an infinitely long subsequence of unobservable events. These requirements are collected in Assumption 1: when it holds, a system will always run and generate an infinitely long observation sequence.

Two fundamental definitions are those of strong detectability and weak detectability originally studied in \([15]\). Strong detectability implies that there exists a positive integer \(k\) such that for all infinite output sequences \(\sigma\) generated by a system, all prefixes of \(\sigma\) of length greater than \(k\) allow reconstructing the current states. Weak detectability implies that there exists a positive integer \(k\) and some infinite output sequence \(\sigma\) generated by a system such that all prefixes of \(\sigma\) of length greater than \(k\) allow reconstructing the current states. It is not difficult to see that weak detectability is strictly weaker than strong detectability. Strong detectability can be verified in polynomial time while weak detectability can be verified in exponential time \([15,12]\) under the usual Assumption 1. In addition, checking weak detectability is PSPACE-complete in the numbers of states and events for FSAs also under Assumption 1 \([21,6]\). For a comprehensive introduction of various notions of strong and weak detectability of FSAs based on Assumption 1, we refer the reader to \([3]\). For a brief introduction to these topics without any assumption, we refer the reader to \([23]\).

The above case that \(\omega\)-languages are considered can well describe long-term behavior of DESs. However, sometimes one needs to consider not only long-term behavior but also short behavior, in this case languages (in this paper languages refer to as a set of words (i.e., finite-length sequences) over an alphabet) are considered. In

---

\(^1\) obtained from a standard finite automaton \([16]\) by removing all accepting states, replacing a unique initial state by a set of initial states, and adding a labeling function...
this case, strong detectability implies that there exists a positive integer \( k \) such that for all finite output sequences \( \sigma \) generated by a system, all prefixes of \( \sigma \) of length greater than \( k \) allow reconstructing the current states, and weak detectability implies that there exists a positive integer \( k \) and some finite output sequence \( \sigma \) generated by a system such that all prefixes of \( \sigma \) of length greater than \( k \) allow reconstructing the current states. In order to distinguish these notions we call strong (resp. weak) detectability in the context of \( \omega \)-languages \( \omega \)-strong (resp. weak) detectability, and similarly call strong (resp. weak) detectability in the context of languages \( * \)-strong (resp. weak) detectability (since languages are subsets of \( \Sigma^* \) for some alphabet \( \Sigma \)). By definition, \( * \)-strong detectability is stronger than \( \omega \)-strong detectability. The polynomial-time verification method given in [12] can be used to verify \( * \)-strong detectability of all FSAs.

In this paper, we study the verification problem and synthesis problem for delayed strong detectability of FSAs. Delays may appear in the observation to a cyber-physical system, because signal transmission through a communication network takes a non-negligible time. For example, when we observed a label generated by a DES at time \( t \), the event that generates the label may have occurred before time \( t \). But in this paper, the notion of delay has a different meaning, i.e., when we observed a sequence \( \sigma \) of labels from the starting of a DES, and we also observed at least a number \( K \) of subsequent labels, can we determine what the state was when \( \sigma \) had just been generated? When \( K = 0 \), it becomes the conventional problem of current-state estimation, i.e., using an observed label sequence to determine the current state. One directly sees that the above “delayed computation” will enforce the possibility of determining the state when \( \sigma \) has just been observed, because we can use more observed information.

Delayed computation may enforce detectability in some cases. Consider the motivating example shown in Fig. 1. One directly sees that this FSA generates only infinite label sequence \( (ab)^\omega \), i.e., the infinite label sequence consisting of infinitely many copies of \( ab \). For every natural number \( n \), when label sequence \( (ab)^n a \) has been observed, then the FSA can be in only state \( s_1 \); but when sequence \( (ab)^{n+1} \) has been observed, the FSA can be in state \( s_0 \) or state \( s_2 \). That is, the FSA is not \( \omega \)-detectable. However, if we consider 1-delayed computation, then the FSA will become \( \omega \)-detectable: when \( (ab)^n a \) has been observed, we know after \( (ab)^n \), the FSA can be only in state \( s_0 \). This yields a more general notion of detectability which we call \( K \)-delayed strong detectability in the context of \( \omega \)-languages (\( \omega \)-\( K \)-delayed strong detectability for short). In particular, strong detectability is exactly 0-delayed strong detectability. We point out that the notion of \( \omega \)-\( K \)-delayed strong detectability can be obtained from the notion of \( (k_1, k_2) \)-detectability originally studied in [13] by replacing “a given \( k_1 \)” by “there exists a natural number \( k_1 \)”.

The contributions of the paper are as follows. (1) We use a novel concurrent-composition method to give polynomial-time algorithms for verifying \( \omega \)-\( K \)-delayed strong detectability and \( * \)-\( K \)-delayed strong detectability of FSAs, where the algorithms apply to all FSAs (Section 3). (2) By using the concurrent-composition method, we also obtain polynomial-time algorithms for verifying \( (k_1, k_2) \)-detectability and \( (k_1, k_2) \)-D-detectability in the contexts of both \( \omega \)-languages and languages for all FSAs, where these notions in the context of \( \omega \)-languages are firstly studied in [13], where the verification algorithms given in [13] apply to FSAs under Assump-
tion 1, as we discuss in Section 4. In Section 5, we use the above method to give polynomial-time synthesis algorithms for enforcing the above notions of delayed strong detectability. Note that in the supervisory control framework, to the best of our knowledge, all known existing algorithms for enforcing delayed strong detectability run in exponential time, see, for example, $\omega$-strong detectability is considered in [14]. $\ast$-$(k_1, k_2)$-detectability is considered in [20], because the usage of a notion of observer (a deterministic finite automaton that is of exponential size of the considered DES) is indispensable. The synthesis algorithm for enforcing a stronger version of $\ast$-strong detectability (obtained by letting the number $K$ in Definition 2 equal to 0, pulling out “$\exists k \in \mathbb{N}$” in Definition 2 and putting “with respect to a given $k \in \mathbb{N}$” before “if”) under the liveness assumption (a little weaker than ii) of Assumption 1) given in [18] is also of exponential-time complexity. The synthesis problem for enforcing $\omega$-strong detectability is also studied in [11] by online sensor activation, where the overall computational complexity is also exponential of the size of the considered DES, based on construction of an automaton square of the size of an observer. In Section 2, we show necessary preliminaries, and in Section 6, we end up this paper with brief discussions on how to use a concurrent-composition method to verify and synthesize diagnosability of FSAs in polynomial time without any assumption, which also improve the related results in the literature.

![Fig. 1](image-url) An FSA, where circles denote states, $t_1, t_2, t_3$ denote events, $a, b$ denote the corresponding labels, a state with an input arrow from nowhere is initial (e.g., $s_0$).

### 2 Preliminaries

An FSA is a sextuple $S = (X, T, X_0, \rightarrow, \Sigma, \ell)$, where $X$ is a finite set of states, $T$ a finite set of events, $X_0 \subset X$ a set of initial states, $\rightarrow \subset X \times T \times X$ a transition relation, $\Sigma$ a finite set of outputs (labels), and $\ell : T \rightarrow \Sigma \cup \{\epsilon\}$ a labeling function, where $\epsilon$ denotes the empty word. The event set $T$ can been rewritten as disjoint union of observable event set $T_o$ and unobservable event set $T_u$, where events of $T_o$ are with label in $\Sigma$, but events of $T_u$ are labeled by $\epsilon$. When an observable event occurs, its label can be observed, but when an unobservable event occurs, nothing can be observed. For an observable event $t \in T$, we say $t$ can be directly observed if $\ell(t)$ differs from $\ell(t')$ for any other $t' \in T$. Labeling function $\ell : T \rightarrow \Sigma \cup \{\epsilon\}$ can be recursively extended to $\ell : T^* \cup T^\omega \rightarrow \Sigma^* \cup \Sigma^\omega$ as $\ell(t_1 t_2 \ldots) = \ell(t_1) \ell(t_2) \ldots$ and $\ell(\epsilon) = \epsilon$. Transitions $x \xrightarrow{t} x'$ with $\ell(t) = \epsilon$ are called $\epsilon$-transitions.
(or unobservable transitions), and other transitions are called observable transitions. The event set \( T \) can also been rewritten as disjoint union of controllable event set \( T_c \) and uncontrollable event set \( T_{uc} \), where controllable events are such that one can disable their occurrences, and uncontrollable events are such that one cannot do that. Analogously, transitions \( x \xleftarrow{t} x' \) with \( t \) being controllable are called controllable, and other transitions are called uncontrollable. With respect to a subset \( X' \subset X \), the semiautomaton is defined by \( S' = \{X', T', \rightarrow \cap (X' \times T \times X') \}, \Sigma, \ell \). Note that for a semiautomaton, initial states are not necessarily assigned. When initial states are assigned, a semiautomaton becomes an FSA.

Next we introduce necessary notions that will be used throughout this paper. Symbols \( \mathbb{N} \) and \( \mathbb{Z}_+ \) denote the sets of natural numbers and positive integers, respectively. For a set \( S \), \( S^+ \) and \( S^\omega \) are used to denote the sets of finite sequences (called words) of elements of \( S \) including the empty word \( \epsilon \) and infinite sequences (called configurations) of elements of \( S \), respectively. As usual, we denote \( S^\omega = S^+ \setminus \{\epsilon\} \). For a word \( s \in S^\omega \), \(|s|\) stands for its length, and we set \(|s'\) = +\( \infty \) for all \( s' \in S^\omega \). For \( s \in S \) and natural number \( k \), \( s^k \) and \( s^\omega \) denote the \( k \)-length word and configuration consisting of copies of \( s \)'s, respectively. For a word (configuration) \( s \in S^\omega (S^\omega) \), a word \( s' \in S^\omega \) is called a prefix of \( s \), denoted as \( s' \subseteq s \), if there exists another word (configuration) \( s'' \in S^\omega (S^\omega) \) such that \( s = s's'' \). For two natural numbers \( i \leq j \), \([i, j] \) denotes the set of all integers between \( i \) and \( j \) including \( i \) and \( j \); and for a set \( S \), \(|S| \) its cardinality and \( 2^S \) its power set.

A state \( x \in X \) is called deadlock if \( (x, t, x') \notin \rightarrow \) for any \( t \in T \) and \( x' \in X \). \( S \) is called deadlock-free if it has no deadlock state. For all \( x, x' \in X \) and \( t \in T \), we also denote \( x \xleftarrow{t} x' \) if \( (x, t, x') \in \rightarrow \). More generally, we denote transitions \( x \xleftarrow{t_1 \ldots t_n} x_1 \), \( x_1 \xleftarrow{t_2} x_2, \ldots, x_{n-1} \xleftarrow{t_n} x_n \) by \( x \xleftarrow{t_1 \ldots t_n} x_n \) for short, where \( n \in \mathbb{Z}_+ \), and call it a transition sequence from \( x \) to \( x_n \) under \( t_1 \ldots t_n \). We say \( S \) generates an event sequence \( s \in T^\omega \) if there is a transition sequence \( x_0 \xleftarrow{t_1 \ldots t_n} x_n \) with \( x_0 \in X_0 \) and \( x \in X \). The set of event sequences generated by \( S \) is denoted by \( T(S) \). We say a state \( x' \in X \) is reachable from a state \( x \in X \) if there exist \( t_1, \ldots, t_n \in T \) such that \( x \xleftarrow{t_1 \ldots t_n} x' \), where \( n \) is a positive integer. We say a subsequence \( X' \) of \( X \) is reachable from a state \( x \in X \) if some state of \( X' \) is reachable from \( x \). Similarly a state \( x \in X \) is reachable from a subsequence \( X' \) of \( X \) if \( x \) is reachable from some state of \( X' \). We call a state \( x \in X \) reachable if either \( x \in X_0 \) or it is reachable from an initial state.

For each \( \sigma \in \Sigma^\omega \), we denote by \( M(S, \sigma) \) the set of states that the system can be in after \( \sigma \) has been observed, i.e., \( M(S, \sigma) := \{x \in X | (\exists x_0 \in \mathbb{X}_0)[(\exists s \in T^\omega)][(\ell(s) = \sigma) \land (x_0 \xleftarrow{s} x)]\} \). In addition, we set \( M(S, \epsilon) := M(S, \epsilon) \cup \mathbb{X}_0 \). When computation delays are considered, the set \( M(S, \sigma) \) can be extended to \( M(S, \sigma_1, \sigma_2) := \{x \in X | (\exists x_0 \in \mathbb{X}_0)[(\exists x' \in X)[(\exists s_1, s_2 \in T^\omega)][(\ell(s_1) = \sigma_1) \land (\ell(s_2) = \sigma_2) \land (x_0 \xleftarrow{s_2} x \xleftarrow{s_1} x')]} \) for all \( \sigma_1 \in \Sigma^\omega \) and \( \sigma_2 \in \Sigma^\omega \). Particularly, \( M(S, \sigma_1, \epsilon) := M(S, \sigma_1) \) for all \( \sigma_1 \in \Sigma^\omega \), \( M(S, \sigma, \sigma_2) := M(S, \sigma, \sigma_2) \cup \{x \in \mathbb{X}_0[(\exists x \in X)[(\exists s \in T^\omega)][(x_0 \xleftarrow{s} x) \land (\ell(s) = \sigma_2)]\} \) for all \( \sigma_2 \in \Sigma^\omega \). \( L(S) \) denotes the language generated by \( S \), i.e., \( L(S) := \{\sigma \in \Sigma^\omega | M(S, \sigma) \neq \emptyset \} \). An infinite event sequence \( t_1 t_2 \ldots t_n \in T^\omega \) is called generated by \( S \) if there exist states \( x_0, x_1, \ldots \in X \) with \( x_0 \in \mathbb{X}_0 \) such that for all \( i \in \mathbb{N} \), \( (x_i, t_{i+1}, x_{i+1}) \in \rightarrow \). We use \( L^\omega(S) \)
to denote the \( \omega \)-language generated by \( S \), i.e., \( \mathcal{L}^\omega(S) := \{ \sigma \in \Sigma^\omega | (\exists t_1 t_2 \ldots \in T^\omega \text{ generated by } S)[(\ell(t_1 t_2 \ldots) = \sigma]\}. 

The following two assumptions are commonly used in detectability studies (cf. \([15,12,13]\)), but are not needed in the current paper.

**Assumption 1** An FSA \( S = (X, T, X_0, \rightarrow, \Sigma, \ell) \) satisfies

(i) \( S \) is deadlock-free,

(ii) \( S \) is prompt, i.e., for every reachable state \( x \in X \) and every nonempty unobservable event sequence \( s \), there exists no transition sequence \( x \rightarrow x \) in \( S \).

### 3 Polynomial-time verification algorithms

We next formulate \( K \)-delayed strong detectability for FSAs, where \( K \in \mathbb{N} \).

**Definition 1** An FSA \( S = (X, T, X_0, \rightarrow, \Sigma, \ell) \) is called \( \omega \)-\( K \)-delayed strongly detectable if

\[(\exists k \in \mathbb{N})(\forall \sigma \in \mathcal{L}^\omega(S))(\forall \sigma_1 \sigma_2 \subseteq \sigma)
\[|((|\sigma_1| \geq k) \land (|\sigma_2| \geq K)) \implies (|M(S, \sigma_1, \sigma_2)| = 1)|\].

**Definition 2** An FSA \( S = (X, T, X_0, \rightarrow, \Sigma, \ell) \) is called \( \ast \)-\( K \)-delayed strongly detectable if

\[(\exists k \in \mathbb{N})(\forall \sigma \in \mathcal{L}(S))(\forall \sigma_1 \sigma_2 \subseteq \sigma)
\[|((|\sigma_1| \geq k) \land (|\sigma_2| \geq K)) \implies (|M(S, \sigma_1, \sigma_2)| = 1)|\].

One directly sees that if \( K = 0 \), then \( K \)-delayed strong detectability reduces to the conventional strong detectability.

We will use a concurrent-composition method to give polynomial-time algorithms for verifying \( K \)-delayed strong detectability for both cases.

In order to show the main results, we need two notions of concurrent composition and observation automaton for an FSA.

Consider an FSA \( S = (X, T, X_0, \rightarrow, \Sigma, \ell) \). We construct its concurrent composition 

\[CC_A(S) = (X', T', X_0', \rightarrow') \tag{1}\]

as follows:

1. \( X' = X \times X \);
2. \( T' = T'_o \cup T'_e \), where \( T'_o = \{(\bar{t}, \bar{l}) | \bar{t}, \bar{l} \in T, \ell(\bar{l}) = \ell(\bar{t}) \in \Sigma\} \), \( T'_e = \{(\bar{t}, \epsilon) | \bar{t} \in T, \ell(\bar{t}) = \epsilon\} \cup \{\epsilon, \bar{l} \in T, \ell(\bar{l}) = \epsilon\} ;
3. \( X'_o = X_0 \times X_0 \); 
4. for all \((\bar{x}_1, \bar{x}'_1), (\bar{x}_2, \bar{x}'_2) \in X'\), \((\bar{t}, \bar{l}) \in T'_o\), \((\bar{t}', \epsilon) \in T'_e\), and \((\epsilon, \bar{l}'') \in T'_o\),
   - \((\bar{x}_1, \bar{x}'_1), (\bar{t}, \bar{l}), (\bar{x}_2, \bar{x}'_2) \in \rightarrow'\) if and only if \((\bar{x}_1, \bar{t}, \bar{x}_2), (\bar{x}'_1, \bar{l}, \bar{x}'_2) \in \rightarrow\);
   - \((\bar{x}_1, \bar{x}'_1), (\epsilon, \bar{l}''), (\bar{x}_2, \bar{x}'_2) \in \rightarrow'\) if and only if \((\bar{x}_1, \bar{l}'', \bar{x}_2) \in \rightarrow, \bar{x}'_1 = \bar{x}'_2, \)
   - \((\bar{x}_1, \bar{x}'_1), (\epsilon, \bar{l}''), (\bar{x}_2, \bar{x}'_2) \in \rightarrow'\) if and only if \((\bar{x}_1 = \bar{x}_2, (\bar{x}'_1, \bar{l}'', \bar{x}'_2) \in \rightarrow. \)
For an event sequence \( s' \in (T')^* \), we use \( s'(L) \) and \( s'(R) \) to denote its left and right components, respectively. Similar notation is applied to states of \( X' \). In addition, for every \( s' \in (T')^* \), we use \( \ell(s') \) to denote \( \ell(s'(L)) \) or \( \ell(s'(R)) \), since \( \ell(s'(L)) = \ell(s'(R)) \). In the above construction, \( CC_A(S) \) aggregates every pair of transition sequences of \( S \) producing the same label sequence. In addition, \( CC_A(S) \) has at most \( |X|^2 \) states and at most \(|X|^2 |2T_o||X| + \sum_{\sigma \in \Sigma} |\ell^{-1}(\sigma)|^2 |X|^2 \) transitions, where the number does not exceed \(|X|^2 |2T_o||X| + |T_o|^2 X^2 \). Hence it takes time \( O(2|X|^3 |T_o| + |X|^4 \sum_{\sigma \in \Sigma} |\ell^{-1}(\sigma)|^2) \) to construct \( CC_A(S) \). For the special case when all observable events can be directly observed studied in [12], the complexity reduces to \( O(2|X|^3 |T_o| + |X|^4 |T_o|) \).

Construct its observation automaton

\[
\text{Obs}(S) = (X, \{\varepsilon, \hat{\varepsilon}\}, X_0, \rightarrow, \{\varepsilon\}, \ell')
\]

in linear time of the size of \( S \), where \( \rightarrow \subset X \times \{\varepsilon, \hat{\varepsilon}\} \times X \), \( \ell'(\varepsilon) = \varepsilon, \ell'(\hat{\varepsilon}) = \hat{\varepsilon} \), for every two states \((x, x') \in X \times \{\varepsilon, \hat{\varepsilon}\} \times X \), \( \ell'(x, x') \mapsto \rightarrow \) if there exists \( t \in T \) such that \((x, t, x') \in \rightarrow \) and \( \ell(t) \neq \varepsilon \); \((x, \varepsilon, x') \in \rightarrow \) if there exists \( t \in T \) such that \((x, t, x') \in \rightarrow \) and \( \ell(t) = \varepsilon \). Here the label function \( \ell' \) is also naturally extended to \( \ell' : \{\varepsilon, \hat{\varepsilon}\}^* \rightarrow \{\varepsilon\}^* \cup \{\hat{\varepsilon}\}^\omega \).

Example 1 An FSA \( S \), its concurrent composition and observation automaton are shown in Fig. 2.

3.1 Verifying \( \omega \)-K-delayed strong detectability

**Theorem 1** The \( \omega \)-K-delayed strong detectability of FSAs can be verified in polynomial time.

**Proof** Consider an FSA \( S = (X, T, X_0, \rightarrow, \Sigma, \ell) \) and the accessible part \( \text{Acc(Obs(CC_A(S)))} = (X', T', X'_0, \rightarrow') \) of the observation automaton of the concurrent composition of \( S \). We claim that \( S \) is not \( \omega \)-K-delayed strongly detectable if and only if in \( \text{Acc(Obs(CC_A(S)))} \),

there exists a transition sequence
for a sketch.

If (3) holds, then in $S$, for every $n \in \mathbb{Z}_+$, there exists a transition sequence

$$x_0'(L) \xrightarrow{s_1'(L)} x_1'(L) \xrightarrow{(s_2'(L))^n} x_1'(L) \xrightarrow{s_2'(L)}$$

such that $|\mathcal{M}(S, \sigma_1, \sigma_2)| > 1$ and at $x_{3+K}'(L)$ there is an infinite-length transition sequence with an infinite-length label sequence, where $\sigma_1 = \ell(s_1'(L))|s_2'(L)| \ell(s_1'(L))$ is of length $\geq n$, $\sigma_2 = \ell(s_1'(L)) \ldots \ell(s_3'(L))$ is of length $\geq K$. Hence $S$ is not $\omega$-$K$-delayed strongly detectable.

Next we show that (3) can be verified in linear time of the size of $\text{Acc(Obs}(CC_A(S)))$. See Fig. 3 for a sketch.

1. Compute $\text{Acc(Obs}(S))$, and then the set $X_{4+K}'$ of states of $\text{Acc(Obs}(S))$ that belong to a cycle with positive-length label sequences.
2. Compute $\text{Acc(Obs}(CC_A(S))) = (X', T', X_0', \rightarrow')$, and then $X_{4+K}' = \{(x, x') \in X'|(\exists x'' \in X_{4+K})[(x', x'') \xrightarrow{\epsilon}\}]$.
3. Compute $X_{4+K}'$, $X_{4+K}'$, ..., $X_4'$ in order as $X_4' = \{(x, x') \in X'|(\exists (x''', x''') \in X_4')|[x, x'] \xrightarrow{(x''', x''')]$.
4. Compute $X_4' = \{(x, x') \in X'|[(x', x'') \xrightarrow{\epsilon}] \wedge ((\exists (x''', x''')) \in X_4')|(\exists s \in \ell(\epsilon)^+ \exists (x, x') \xrightarrow{(x''', x''')]$.
5. Compute $X_2' = \{(x, x') \in X'|[x, x'] \xrightarrow{\epsilon}] \wedge ((\exists s \in \ell(\epsilon)^+ \{\epsilon\}^+ \exists (x, x') \xrightarrow{(x, x')}$.
6. $X_2' \neq \emptyset$ if and only if $S$ is not $\omega$-$K$-delayed strongly detectable.

Step 1 can be implemented in linear time of the size of $S$. Compute all strongly connected components of $\text{Acc(Obs}(S))$, which can be done by using the well-known depth-first search in linear time of $S$. By definition, if a strongly connected component contains a transition, then it contains a cycle consisting of all vertices and all transitions (repeated of states and transitions permitted) in the component. Then a
state $x$ belongs to a cycle with a nonempty label sequence if and only if there is an observable transition in the strongly connected component to which $x$ belongs. And hence $X_{4+K}$ consists of all states of all strongly connected components that contain at least one observable transition. Hence Step 1 can be finished in time at most twice of the size of $S$.

In Step 2, firstly compute $X'_{4+K} = \{ x \in \text{Acc(Obs(S))} \vert \text{either } x \in X_{4+K} \text{ or } X_{4+K} \text{ is reachable from } x \}$, and then $X''_{3+K} = \{ (x,x') \in X' \vert x \in X'_{4+K} \}$. Hence Step 2 can be finished in time that does not exceed twice of the size of $\text{Acc(Obs(CC_A(S)))}$.

In Step 3, taking $X''_{2+K}$ for example, firstly compute $X'''_{3+K} = \{ (x,x') \in X' \vert x \text{ is reachable from } (x,x') \text{ through } \epsilon\text{-transitions} \} \cup X'$, secondly compute $X''''_{3+K} = \{ (x,x') \in X' \vert \text{ there is an observable transition from } (x,x') \text{ to some state of } X''_{3+K} \}$. Then $X''''_{3+K} = X''_{2+K}$. Hence computing $X''_{2+K}$ takes time at most the size of $\text{Acc(Obs(CC_A(S)))}$. Hence Step 3 can be finished in time at most $K-1$ times of the size of $\text{Acc(Obs(CC_A(S)))}$.

Similarly, Step 4 can be finished in time at most the size of $\text{Acc(Obs(CC_A(S)))}$.

Step 5 can be implemented in time at most the size of $\text{Acc(Obs(CC_A(S)))}$ by the argument in Step 1.

Based on the above argument, the overall computational cost of verifying $\omega$-$K$-delayed strong detectability does not exceed twice of the size of $S$ plus $K + 3$ times of the size of $\text{Acc(Obs(CC_A(S)))}$. Hence it takes time $O((K + 3)(2|X|^2|T_{\epsilon}| + |X|^4 \sum_{\sigma \in \Sigma} |f^{-1}(\sigma)|^2))$ to verify $\omega$-$K$-delayed strong detectability. For the special case when all observable events can be directly observed studied in [12], the complexity reduces to $O((K + 3)(2|X|^2|T_{\epsilon}| + |X|^4|T_{\epsilon}|))$.

$\square$

Example 2 Recall the FSA $S$ in the left part of Fig. 2. We first verify its $\omega$-strong detectability. Following the procedure in the proof of Theorem 1, we have $X_4 = \{ s_0, s_1 \}$, $X'_3 = \{ (s_1, s_2), (s_2, s_1) \}$, $X'_2 = \{ (s_0, s_0) \} \neq \emptyset$, then $S$ is not $\omega$-strongly detectable.
Next we verify its ω-1-delayed strong detectability. Similarly, we have \(X_5 = \{s_0, s_1\}, X_4 = \{(s_0, s_0), (s_1, s_1), (s_1, s_2), (s_2, s_1)\}\), \(X_3 = \emptyset, X_2 = \emptyset\), then \(S\) is ω-1-delayed strongly detectable.

**Remark 1** By Example 2, one sees that ω-strong detectability is not equivalent to ω-1-delayed strong detectability. Hence ω-strong detectability is strictly stronger than ω-K-delayed strong detectability for some positive integer \(K\).

Next by using the verification method in the proof of Theorem 1, an upper bound for \(K\) in the notion of ω-K-delayed strong detectability can be easily obtained.

**Corollary 1** If an FSA is ω-K-delayed strongly detectable for some \(K \in \mathbb{N}\), then it is also ω-\(|X|^2\)-delayed strongly detectable, where \(X\) is the state set of the FSA.

**Proof** Assume that the FSA is not ω-\(|X|^2\)-delayed strongly detectable, then by the proof of Theorem 1, we have the corresponding sets \(X_1, X_2, \ldots, X_{\mathbb{N}}\) are all nonempty (see Fig. 3 for a sketch). On the other hand, since there exist at most \(|X|^2\) distinct subsets of \(X\) with cardinality 2, one has at least two of \(X_0', X_1', \ldots, X_{\mathbb{N}}'\) are the same. Hence for each \(K \geq |X|^2\), the FSA is not ω-K-delayed strongly detectable.

**Remark 2** Actually, this upper bound has been given in [13] under Assumption 1. Hence the algorithm for verifying ω-(\(k_1, k_2\))-detectability can be used to verify ω-k_2-delayed strong detectability by verifying ω-(\(|X|^2, k_2\))-detectability under Assumption 1, since by the upper bound one sees that ω-k_2-delayed strong detectability is equivalent to ω-(\(|X|^2, k_2\))-detectability.

**Remark 3** By Corollary 1, the computational cost of verifying ω-K-delayed strong detectability can be reduced to twice of the size of \(S\) plus \((\min\{K, |X|^2\} + 3)\) times of the size of \(\text{Acc(Obs(CC}_{A}(S))}\).

### 3.2 Verifying \(\ast\)-K-delayed strong detectability

**Theorem 2** The \(\ast\)-K-delayed strong detectability of FSAs can be verified in polynomial time.

**Proof** Consider an FSA \(S = (X, T, X_0, \rightarrow, \Sigma, \ell)\) and \(\text{Acc(Obs(CC}_{A}(S))}\) = \((X', T', X_0', \rightarrow', \ell')\). Similarly to ω-K-delayed strong detectability (Theorem 1), we can prove \(S\) is not \(\ast\)-K-delayed strongly detectable if and only if in \(\text{Acc(Obs(CC}_{A}(S))}\),

there exists a transition sequence

\[
\begin{align}
&x_0' \overset{s_1'}{\longrightarrow} x_1' \overset{s_2'}{\longrightarrow} \cdots \overset{s_{3+K}'}{\longrightarrow} x_{3+K}' \\
&x_0' \in X_0'; x_1', \ldots, x_{3+K}' \in X'; s_1', \ldots, s_{3+K}' \in (T')^*; \\
&x_1' = x_2'; \ell(s_2') \in \Sigma^+; x_3'(L) \neq x_3'(R); \\
&|\ell(s_i')| = 1, i \in [4, 3 + K].
\end{align}
\]

(4a) (4b) (4c) (4d)
Next we show that (4) can be verified in linear time of the size of $\text{Acc}(\text{Obs}(CC_A(S)))$. See Fig. 4 for a sketch. We verify (4) by computing as in Fig. 4

$$X'_3 + K = X'_1,$$
$$X'_2 + K = \{(x, x') \in X' \mid (\exists (x'', x''') \in X'_3 + K)$$
$$\quad \exists s \in \tilde{\epsilon}[x^*)[(x, x') \xrightarrow{\sigma} (x'', x''')]],$$
$$\vdots$$
$$X'_1 = \{(x, x') \in X' \mid (\exists (x'', x''') \in X'_4)$$
$$\quad \exists s \in \tilde{\epsilon}[x^*)[(x, x') \xrightarrow{\sigma} (x'', x''')]],$$
$$X'_0 = \{(x, x') \mid (X'_3 is reachable from (x, x') \land$$
$$\quad (\exists s \in \tilde{\epsilon}[x^*)[(x, x') \xrightarrow{\sigma} (x, x')])\},$$

$X'_0 \neq \emptyset$ if and only if $S$ is not $-K$-delayed strongly detectable.

By similar analysis to Theorem 1, one has the overall computational cost of verifying $-K$-delayed strong detectability does not exceed $K + 2$ times of the size of $\text{Acc}(\text{Obs}(CC_A(S)))$. Hence it takes time $O((K + 2)(2|X|^3|T_\epsilon| + |X|^4 \sum_{\sigma \in \Sigma} (\tilde{\epsilon}^{-1}(\sigma))^2))$ to verify $-K$-delayed strong detectability.

\[\square\]

**Example 3** Recall the FSA $S$ in the left part of Fig. 2. We first verify its $s$-strong detectability. Following the procedure in the proof of Theorem 2, we have $X'_3 = \{(s_1, s_2), (s_2, s_1)\}, X'_2 = \{(s_0, s_0)\} \neq \emptyset$, then $S$ is not $s$-strongly detectable.

Next we verify its $s$-1-delayed strong detectability. Similarly, we have $X'_4 = \{(s_0, s_0), (s_1, s_1), (s_1, s_2), (s_2, s_1), (s_2, s_2)\}, X'_3 = \emptyset, X'_2 = \emptyset$, then $S$ is $s$-1-delayed strongly detectable.

Further analysis to $s$-K-delayed strong detectability can also be done as for $\omega$-$K$-delayed strongly detectable. Here we only state the corresponding results.

**Corollary 2** If an FSA is $s$-$K$-delayed strongly detectable for some $K \in \mathbb{N}$, then it is also $s$-$|X|^2$-delayed strongly detectable, where $X$ is the state set of the FSA.

**Remark 4** By Corollary 2, the computational cost of verifying $s$-$K$-delayed strong detectability can be reduced to $(2 \min\{K, |X|^2\} + 2)$ times of the size of $\text{Acc}(\text{Obs}(CC_A(S)))$. 

\[\text{Fig. 4 Sketch for verifying (4), i.e., negation of $-K$-delayed strong detectability.}\]
In [13], two notions of \((k_1, k_2)\)-detectability and \((k_1, k_2)\)-D-detectability for FSAs in the context of \(\omega\)-languages for some \(k_1, k_2 \in \mathbb{N}\) are characterized, and polynomial-time verification algorithms for these notions under Assumption 1 are designed. As an application of our results given in Section 3, we give polynomial-time verification algorithms for \((k_1, k_2)\)-detectability and \((k_1, k_2)\)-D-detectability in the contexts of \(\omega\)-languages and languages without any assumption.

**Definition 3** An FSA \(S = (X, T, X_0, \rightarrow, \Sigma, \ell)\) is called \(\omega-(k_1, k_2)\)-detectable if
\[
(\forall \sigma \in \mathcal{L}(S))(\forall \sigma_1, \sigma_2 \subseteq \sigma)
[\((|\sigma_1| \geq k_1) \land (|\sigma_2| \geq k_2)) \implies (|\mathcal{M}(S, \sigma_1, \sigma_2)| = 1)].
\]

**Definition 4** An FSA \(S = (X, T, X_0, \rightarrow, \Sigma, \ell)\) is called \(\ast-(k_1, k_2)\)-detectable if
\[
(\forall \sigma \in \mathcal{L}(S))(\forall \sigma_1, \sigma_2 \subseteq \sigma)
[\((|\sigma_1| \geq k_1) \land (|\sigma_2| \geq k_2)) \implies (|\mathcal{M}(S, \sigma_1, \sigma_2)| = 1)].
\]

Consider a specification \(X_{\text{spec}} \subseteq X \times X\), where each state pair of \(X_{\text{spec}}\) is crucial and the two states of such pairs must be distinguished.

**Definition 5** An FSA \(S = (X, T, X_0, \rightarrow, \Sigma, \ell)\) is called \(\omega-(k_1, k_2)\)-D-detectable with respect to specification \(X_{\text{spec}} \subseteq X \times X\) if
\[
(\forall \sigma \in \mathcal{L}(S))(\forall \sigma_1, \sigma_2 \subseteq \sigma)
[\((|\sigma_1| \geq k_1) \land (|\sigma_2| \geq k_2)) \implies (\mathcal{M}(S, \sigma_1, \sigma_2) \times \mathcal{M}(S, \sigma_1, \sigma_2) \cap X_{\text{spec}} = \emptyset)].
\]

**Definition 6** An FSA \(S = (X, T, X_0, \rightarrow, \Sigma, \ell)\) is called \(\ast-(k_1, k_2)\)-D-detectable with respect to specification \(X_{\text{spec}} \subseteq X \times X\) if
\[
(\forall \sigma \in \mathcal{L}(S))(\forall \sigma_1, \sigma_2 \subseteq \sigma)
[\((|\sigma_1| \geq k_1) \land (|\sigma_2| \geq k_2)) \implies (\mathcal{M}(S, \sigma_1, \sigma_2) \times \mathcal{M}(S, \sigma_1, \sigma_2) \cap X_{\text{spec}} = \emptyset)].
\]

One directly sees that \((k_1, k_2)\)-detectability is stronger than \(k_2\)-delayed strong detectability in the contexts of \(\omega\)-languages and languages. The former is strictly stronger than the latter. Consider the FSA \(S\) in Fig. 5, one directly sees that the FSA is not \(\omega-(0, 0)\)-detectable (by \(\mathcal{M}(S, \epsilon, \epsilon) = \mathcal{M}(S, \epsilon) = \{(s_0, s'_0)\}\), but is \(\omega\)-0-delayed strongly detectable. One also sees that \((k_1, k_2)\)-detectability is stronger than \((k_1, k_2)\)-D-detectability. If we choose \(X_{\text{spec}} = \{(x, x') \in X \times X | x \neq x'\}\), then \((k_1, k_2)\)-D-detectability reduces to \((k_1, k_2)\)-detectability.
4.1 Verifying $\omega$-$(k_1, k_2)$-detectability and $\omega$-$(k_1, k_2)$-D-detectability

Next by using a procedure similar to the proof of Theorem 1, we give polynomial-time verification algorithms for $(k_1, k_2)$-detectability and $(k_1, k_2)$-D-detectability in the context of $\omega$-languages. These results strengthen the corresponding results given in [13] under Assumption 1.

**Theorem 3** The $\omega$-$(k_1, k_2)$-detectability of FSAs can be verified in polynomial time.

**Proof** Consider an FSA $S = (X, T, X_0, \rightarrow, \Sigma, \ell)$ and $\text{Acc}(\text{Obs}(\text{CC}_A(S))) = (X', T', X_0', \rightarrow')$. Similarly to the proof of Theorem 1, we can prove that $S$ is not $\omega$-$(k_1, k_2)$-detectable if and only if in $\text{Acc}(\text{Obs}(\text{CC}_A(S)))$,

there exists a transition sequence

$x_0' \xrightarrow{s_1'} x_1' \xrightarrow{s_2'} \cdots \xrightarrow{s_{k_1+k_2}'} x_{k_1+k_2}'$ satisfying

$(5a)$

$x_0' \in X_0'; x_1', \ldots, x_{k_1+k_2}' \in X'; s_1', \ldots, s_{k_1+k_2}' \in (T')^*$;

$(5b)$

$\ell(s_1'), \ldots, \ell(s_{k_1}') \in \Sigma^+; x_{k_1}'(L) \neq x_{k_1}'(R);

$(5c)$

$|\ell(s_1')| = \cdots = |\ell(s_{k_1+k_2}')| = 1$;

$(5d)$

and in $S$, there exists a cycle with a nonempty label sequence reachable from $x_{k_1+k_2}'(L)$.

$(5e)$

Next we show that $(5)$ can be verified in linear time of the size of $\text{Acc}(\text{Obs}(\text{CC}_A(S)))$. We verify $(5)$ as in Fig. 6.

1. Compute $\text{Acc}(\text{Obs}(S))$, and then the set $X_{1+k_1+k_2}$ of states of $\text{Acc}(\text{Obs}(S))$ that belong to a cycle with positive-length label sequences.
2. Compute $\text{Acc}(\text{Obs}(\text{CC}_A(S))) = (X', T', X_0, \rightarrow')$, and then $X'_{k_1+k_2} = \{(x, x') \in X'|\exists x'' \in X_{1+k_1+k_2}\}$. If $x'$ is reachable from $x$.

3. Compute $X'_{k_1+k_2-1}, \ldots, X'_{k_1+1}$ in order as $X'_i = \{(x, x') \in X'|\exists (x'', x''') \in X_{k_1+1} (\exists s \in \hat{\varepsilon}(x'), x' \xrightarrow{\hat{\varepsilon}(x')} (x'', x'''))\}.

4. Compute $X'_{k_1} = \{(x, x') \in X'|[x' \neq x''] \land [(\exists (x'', x''') \in X_{k_1+1}) (\exists s \in \hat{\varepsilon}(x'), x' \xrightarrow{\hat{\varepsilon}(x')} (x'', x'''))]\}.

5. Compute $X'_{k_1-1}, \ldots, X'_i$ in order as $X'_i = \{(x, x') \in X'|[\exists (x'', x''') \in X_{k_1+1}] (\exists s \in \hat{\varepsilon}(x'), x' \xrightarrow{\hat{\varepsilon}(x')} (x'', x'''))\}.

6. Compute $X'_0 = \{(x, x') \in X_0|\exists (x'', x''') \in X'_1 (\exists s \in \hat{\varepsilon}(x'), x' \xrightarrow{\hat{\varepsilon}(x')} (x'', x'''))\}$. If $x_0 \neq 0$ if and only if $S$ is not $\omega$-($k_1, k_2$)-detectable.

By similar analysis to Theorem 1, one has the overall computational cost of verifying $\omega$-($k_1, k_2$)-detectability does not exceed twice of the size of $S$ plus $(k_1 + k_2 + 2)$ times the size of $\text{Acc}(\text{Obs}(\text{CC}_A(S)))$. Hence it takes time $O((k_1 + k_2 + 2)|X|T, |X|^4 \sum_{\sigma \in \Sigma} |(\varepsilon^{-1}(\sigma))^2|)$ to verify $\omega$-($k_1, k_2$)-detectability.

We have shown that $\omega$-($k_1, k_2$)-detectability is strictly stronger than $\omega$-$k_2$-delayed strongly detectable. Conversely, one also sees if an FSA is $\omega$-$K$-delayed strongly detectable then it is $\omega$-($k, K$)-detectable for some $k \in \mathbb{N}$. In more detail, we have the following proposition.

**Proposition 1** If an FSA $S$ is $\omega$-$K$-delayed strongly detectable then it is $\omega$-($k, K$)-detectable for some $k \leq |X|^2$, where $X$ is the state set of $S$.

**Proof** Suppose on the contrary that $S$ is not $\omega$-($k, K$)-detectable, then by definition it is not $\omega$-($k, K$)-detectable for any $k < |X|^2$; and by the proof of Theorem 3, there is a sequence (5a) with $k_1 = |X|^2$ and $k_2 = K$. Since there exist at most $|X|^2$ distinct subsets of $X$ with cardinality 2, at least two of $x'_1, \ldots, x'_{k_1}$ are the same. Hence $S$ is not $\omega$-($k'_1, K$)-detectable for any $k'_1 \geq |X|^2$. Thus $S$ is not $\omega$-$K$-delayed strongly detectable.

**Theorem 4** The $\omega$-($k_1, k_2$)-D-detectability of FSAs with respect to a specification can be verified in polynomial time.

**Proof** Consider an FSA $S = (X, T, X_0, \rightarrow, \Sigma, \ell), \text{Acc}(\text{Obs}(\text{CC}_A(S))) = (X', T', X_0, \rightarrow')$, and a specification $X_{\text{spec}} \subset X \times X$. Similar to the proof of Theorem 3, we can prove that $S$ is not $\omega$-($k_1, k_2$)-D-detectable with respect to $X_{\text{spec}}$ if and only if in $\text{Acc}(\text{Obs}(\text{CC}_A(S)))$.

there exists a transition sequence

\[
x'_0 \xrightarrow{s'_1} x'_1 \xrightarrow{s'_2} \ldots \xrightarrow{s'_{k_1+k_2}} x'_{k_1+k_2},
\]

satisfying

\[
x'_0 \in X_0; x'_1, \ldots, x'_{k_1+k_2} \in X'; s'_1, \ldots, s'_{k_1+k_2} \in (T')^*;
\]

\[
\ell(s'_1), \ldots, \ell(s'_{k_1}) \in \Sigma^+; x'_{k_1}(L) \neq x'_{k_1}(R);
\]

\[
|\ell(s'_{k_1})| = \cdots = |\ell(s'_{k_1+k_2})| = 1;
\]

\[
(x'_{k_1}(L), x'_{k_1}(R)) \lor (x'_{k_1}(R), x'_{k_1}(L)) \in X_{\text{spec}};
\]

and in $S$, there exists a cycle with a nonempty label...
sequence reachable from \( x_{k_1+k_2}'(L) \).

By using almost the same procedure to that in the proof of Theorem 3 (only by replacing \( X_{k_1}' \) to \( X_{k_1}' \cap X_{\text{spec}} \)), (6) can be checked with the same complexity as for verifying (5). \( \square \)

4.2 Verifying \( \ast(k_1, k_2) \)-detectability and \( \ast(k_1, k_2) \)-D-detectability

Similarly to \( \omega(k_1, k_2) \)-detectability and \( \omega(k_1, k_2) \)-D-detectability, the results for verifying \( \ast(k_1, k_2) \)-detectability and \( \ast(k_1, k_2) \)-D-detectability are shown as follows.

**Theorem 5** The \( \ast(k_1, k_2) \)-detectability of FSAs can be verified in polynomial time.

**Proof** Consider an FSA \( S = (X, T, X_0, \to, \Sigma, \ell) \) and \( \text{Acc(Obs(CC_A(S)))} = (X', T', X_0', \to') \). Similarly to the proof of Theorem 3, we can prove that \( S \) is not \( \ast(k_1, k_2) \)-detectable if and only if in \( \text{Acc(Obs(CC_A(S)))} \),

there exists a transition sequence \( x_0' \xrightarrow{s'_1} x_1' \xrightarrow{s'_2} \cdots \xrightarrow{s'_{k_1+k_2}} x_{k_1+k_2}' \) satisfying

\[
\begin{align*}
&x_0' \in X_0'; x_1', \ldots, x_{k_1+k_2}' \in X'; s_1', \ldots, s_{k_1+k_2}' \in (T')^*; \\
&\ell(s_1'), \ldots, \ell(s_{k_1+k_2}') \in \Sigma^+; x_{k_1}'(L) \neq x_{k_1}'(R); \\
&|\ell(s_{k_1+k_2}')| = \cdots = |\ell(s_{k_1+k_2}')| = 1.
\end{align*}
\]

Next we show that (7) can be verified in linear time of the size of \( \text{Acc(Obs(CC_A(S)))} \). We verify (7) as in Fig. 7.

1. Compute \( \text{Acc(Obs(CC_A(S)))} = (X', T', X_0', \to') \), and \( X_{k_1+k_2}' = X' \).
2. Compute \( X_1'_{k_1+k_2-1}, \ldots, X_1'_{k_1+1} \) in order as
   \[
   X_1' = \{(x, x') \in X'| (\exists s \in \ell(x)\star)[(x, x') \xrightarrow{s} (x'', x''')]) \}\.
   \]
3. Compute \( X_1'_{k_1} = \{(x, x') \in X'| x' \neq x'' \wedge ((\exists (x'', x''') \in X_1'_{k_1+1} ) \exists s \in \ell(x)\star)[(x, x') \xrightarrow{s} (x'', x''')]) \}.

Fig. 7 Sketch for verifying (7), i.e., negation of \( \ast(k_1, k_2) \)-detectability.
4. Compute \( X'_{k_1-1}, \ldots, X'_1 \) in order as \( X'_1 = \{(x, x') \in X'(\exists(x'', x''' \in X'_{k_1+1})(\exists s \in \hat{\epsilon}(\hat{e})^+)(x, x') \overset{\hat{s}}{\rightarrow} (x'', x''')\} \).

5. Compute \( X_0 = \{(x, x') \in X_0'(\exists(x'', x''' \in X'_1)(\exists s \in \{\hat{e}, \hat{e}^+\}^+)(x, x') \overset{\hat{s}}{\rightarrow} (x'', x''')\} \). \( X_0 \neq \emptyset \) if and only if \( S \) is not \( -k_1, k_2\)-detectable.

By similar analysis to Theorem 3, one has the overall computational cost of verifying \( -k_1, k_2\)-detectability does not exceed \((k_1 + k_2 + 1)\) times of the size of \( \text{Acc}(\text{Obs}(\text{CC}_A(S))) \). Hence it takes time \( O((k_1 + k_2 + 1)(2|X|^3|T_e| + |X|^4 \sum_{\sigma \in \Sigma} (|\ell^{-1}(\sigma)|^2)) \) to verify \( -k_1, k_2\)-detectability.

**Theorem 6** The \( -k_1, k_2\)-D-detectability of FSAs with respect to a specification can be verified in polynomial time.

**Proof** Consider an FSA \( S = (X, T, X_0, \rightarrow, \Sigma, \ell) \), \( \text{Acc}(\text{Obs}(\text{CC}_A(S))) = (X', T', X_0', \rightarrow') \), and a specification \( X_{\text{spec}} \subset X \times X \). Similar to the proof of Theorem 5, we can prove that \( S \) is not \( -k_1, k_2\)-D-detectable with respect to \( X_{\text{spec}} \) if and only if in \( \text{Acc}(\text{Obs}(\text{CC}_A(S))) \). There exists a transition sequence

\[
\begin{align*}
  x_0' & \xrightarrow{s_1'} x_1' \xrightarrow{s_2'} \cdots \xrightarrow{s_{k_1+k_2}'} x_{k_1+k_2}' \text{ satisfying} & \tag{8a} \\
  x_0' \in X_0', x_1', \ldots, x_{k_1+k_2}' \in X' ; s_1', \ldots, s_{k_1+k_2}' \in (T')^* ; & \tag{8b} \\
  \ell(s_1') \in \Sigma^* \text{; } x_{k_1}'(L) \neq x_{k_1}'(R) ; & \tag{8c} \\
  |\ell(s_{k_1+k_2}')| = 1 ; & \tag{8d} \\
  (x_{k_1}'(L), x_{k_1}'(R)) \text{ or } (x_{k_1}'(R), x_{k_1}'(L)) \in X_{\text{spec}} . & \tag{8e}
\end{align*}
\]

By using almost the same procedure to that in the proof of Theorem 5 (only by replacing \( X'_{k_1} \) to \( X_{k_1}' \cap X_{\text{spec}} \)), (8) can be checked with the same complexity as for verifying (7).

**5 Polynomial-time synthesis algorithms**

In this section, we study the synthesis problems for enforcing variant notions of delayed strong detectability of FSAs by using the verification methods given in the previous sections. That is, given an undetectable FSA, whether one can make it detectable by disabling several controllable transitions, and how to compute a set of controllable transitions to disable if the answer is yes. We only need to study \( -k_1, k_2\)-detectability and \( -k_1, k_2\)-detectability. Other types of delayed strong detectability can be dealt with similarly.

A popular synthesis method in the supervisory control framework ([9, 21]) is as follows: A partially observed supervisor is a function (in this framework, labeling function \( \ell \) is assumed by default to satisfy \( \ell(t) = t \) for all \( t \in T_o \), hence we assume \( T_o = \Sigma \)).

\[ \text{Sup} : \mathcal{L}(S) \to \Gamma, \]
where  \( \Gamma = \{ \gamma \subset T | T_{uc} \subset \gamma \} \) denotes the set of control decisions, i.e., for an observed label sequence \( s \in L(S) \), \( Sup(s) \) is the set of events enabled upon the observation of \( s \). Then the language generated by the closed-loop system \( L(Sup/S) \) is defined recursively by:

- \( \varepsilon \in L(Sup/S) \);
- for all \( s \in \Sigma^* \), \( \sigma \in \Sigma \), \( s\sigma \in L(Sup/S) \) if and only if \( s \in L(Sup/S) \), \( \sigma \in L(S(Sup/S)) \), and \( s\sigma \in L(S) \).

Given an FSA \( S \) that is not \( -(k_1, k_2) \)-detectable, the synthesis method for enforcing \( -(k_1, k_2) \)-detectability of \( S \) given in [20] with assumption \( T_c \subset T_o \) is firstly to construct a \( (k_1, k_2) \)-observer that generalizes the conventional observer (cf. [15]) by considering delayed information, secondly to use the \( (k_1, k_2) \)-observer to compute a supervisor \( Sup \) as an FSA that enforces \( -(k_1, k_2) \)-detectability of \( S \) if \( Sup \) exists, and finally to compute the parallel composition of \( Sup \) and \( S \), then the parallel composition is \( -(k_1, k_2) \)-detectable. Note that the conventional observer is exponential of the size of \( S \), the \( (k_1, k_2) \)-observer is even larger since its states contain two more components that record information on observed label sequences before (related to \( k_1 \)) and after (related to \( k_2 \)) the time when state estimate is done.

We next give polynomial-time synthesis algorithms for enforcing delayed strong detectability of FSAs by using the verification methods given in Section 3 and Section 4. The synthesis problem considered in our paper is not totally the same as the one in the supervisory control framework [11, 14, 18, 20]. For the supervisory control framework, the synthesis problem is to disable controllable events according to observed labeling sequences, but does not directly depend on structure of a DES. It is somehow the output feedback control. However, our synthesis problem is to directly change part of structure of the DES, i.e., to disable some controllable transitions. Both synthesis processes can be carried out before the DES starts to run.

For an FSA \( S \), the set of its controllable transitions is denoted by \( T^c \).

**Problem 1 (SynEnfDSD)**

1. Given a non-\( \omega -(k_1, k_2) \)-detectable (resp. non-\( -(k_1, k_2) \)-detectable) FSA \( S \), determine whether there is a subset \( T_c \) of controllable transitions such that the new FSA \( S_1 \) obtained from \( S \) by disabling all transitions of \( T_c \) is \( \omega -(k_1, k_2) \)-detectable (resp. \( -(k_1, k_2) \)-detectable).

2. If \( T_c \) exists, how to compute \( T_c \).

We firstly give a theorem that solves Item 1 of Problem 1.

**Theorem 7** Consider a non-\( \omega -(k_1, k_2) \)-detectable (resp. non-\( -(k_1, k_2) \)-detectable) FSA \( S \) and two subsets \( T_c \) and \( T'_c \) of its controllable transitions with \( T_c \subset T'_c \). If \( S_1 \) \( \omega -(k_1, k_2) \)-detectable (resp. \( -(k_1, k_2) \)-detectable), then \( S_{1 \setminus T_c} \) is also \( \omega -(k_1, k_2) \)-detectable (resp. \( -(k_1, k_2) \)-detectable).

**Proof** Assume that \( S_{1 \setminus T'_c} \) is not \( \omega -(k_1, k_2) \)-detectable. Then by Theorem 3, in \( Acc(Obs(CC_A(S_{1 \setminus T'_c})) \) (5) holds. Since every transition of \( S_{1 \setminus T'_c} \) is also a transition of \( S_1 \), one has every transition of \( Acc(Obs(CC_A(S_{1 \setminus T'_c})) \) is also a transition of
without any redundant elements, which means that $S_{1, T}$ is not $\omega$-(k₁, k₂)-detectable either.

The case for $\ast$-(k₁, k₂)-detectability can be proved analogously. \hfill \Box

**Corollary 3** Consider a non-$\omega$-(k₁, k₂)-detectable (resp. non-$\ast$-(k₁, k₂)-detectable) FSA $S$. If $S_{1, T}$ is not $\omega$-(k₁, k₂)-detectable (resp. $\ast$-(k₁, k₂)-detectable), then Item 1 of Problem 1 has no solution.

By Corollary 3, in order to solve Item 1 of Problem 1, we could check whether $S_{1, T}$ is detectable. Hence Item 1 can be solved in polynomial time. Next, we assume $S_{1, T}$ is detectable, and study how to compute a subset $T_{c} \subset T_{c}^{S}$ as small as possible such that $S_{1, T_{c}}$ is detectable. To this end, taking $\ast$-(k₁, k₂)-detectability for example, we need to compute exactly the corresponding $X'_{k₁+k₂}, X'_{k₁+k₂-1}, \ldots, X'_{1}, X'_{0}$ for $S$ as in the proof of Theorem 3 without any redundant elements, which means that from each element of $X'_{0}$ there is a transition sequence to some element of $X'_{k₁+k₂}$ through $X'_{k₁}, \ldots, X'_{1}, X'_{0}$ one by one. After that, we try to find as few as possible controllable transitions to disable in order to cut off all such transition sequences from $X'_{0}$ to $X'_{k₁+k₂}$ through $X'_{k₁}, \ldots, X'_{k₁+k₂-1}$. Then the obtained FSA becomes detectable. The details are shown in the following subsections.

### 5.1 Synthesizing $\omega$-(k₁, k₂)-detectability

We are given a non-$\omega$-(k₁, k₂)-detectable FSA $S = (X, T, X_{0}, \rightarrow, \Sigma, \ell)$ and compute $\text{Acc}(\text{CC}_{A}(S)) = (X', T', X'_{0} \rightarrow')$. Then in $\text{Acc}(\text{CC}_{A}(S))$, (5) holds. The central idea of synthesizing its $\omega$-(k₁, k₂)-detectability is to cut off all sequences shown in (5). In detail, given $k₁, k₂ \in \mathbb{N}$, we do computations based on $\text{Acc}(\text{CC}_{A}(S))$ and $\text{Acc}(S)$ as follows:

1. Compute semiautomaton $S'_{\text{min}\{1, k₁\}}$:
   - $k₁ = 0$: $S'_{0} := \text{Acc}(\text{CC}_{A}(S))$, mark all states $(x, x')$ of $S'_{0}$ with $x \neq x'$.
   - $k₁ = 1$: Compute all transition sequences from all initial states of $X'_{0}$ under an observable event sequence ended with a state $(x, x') \in X'$ satisfying $x \neq x'$, mark $(x, x')$, where observable event sequences are event sequences that contain at least one observable event, i.e., in $(T')^{+} \setminus (T')^{*}$.
   - $k₁ > 1$: Compute all transition sequences from all initial states of $X'_{0}$ under an observable event sequence. Mark the terminal states of these transition sequences.

2. in case of $k₁ > 2$: Compute semiautomata $S'_{1}, \ldots, S'_{k₁-1}$ in order as follows: for all $i \in [2, k₁ - 1]$, from each marked state of $S'_{i-1}$, outside $S'_{i} \cup \cdots \cup S'_{k₁-1}$, compute all transition sequences under an event sequence initialized with an observable event, i.e., an event sequence in $\ell(\varepsilon, \ell)^{*}$. Mark the terminal states of these transition sequences. (For each possible i, states of $S'_{i}$ would be renamed in order to distinguish them from states of $S'_{i} \cup \cdots \cup S'_{k₁-1}$. The same would be done as follows in order to make $S'_{\text{min}\{1, k₁\}}, \ldots, S'_{k₁+k₂}$ have pairwise disjoint sets of states.)
3. in case of $k_1 > 1$: Compute semiautomaton $S'_{k_1-1}$: from each marked state of $S'_{k_1-1}$, outside $S'_1 \cup \cdots \cup S'_{k_1-1}$, compute all transition sequences under an event sequence initialized with an observable event, where these transition sequences are also ended with a state $(x, x')$ with $x \neq x'$ in $X$, mark $(x, x')$.

4. in case of $k_2 > 0$: Compute semiautomata $S'_{k_1+1}, \ldots, S'_{k_1+k_2}$ in order as follows: for all $i \in [k_1 + 1, k_1 + k_2]$, from each marked state of $S'_{k_1}, \ldots, S'_{k_1+k_2}$, compute all transition sequences under an event sequence initialized with an observable event followed by only unobservable events, i.e., event sequences in $\epsilon \{\epsilon\}^\ast$. Mark all its states.

5. Compute $X_{1+k_1+k_2} = \{x \in X | (\exists x' \in X) [\text{either } (x, x') \text{ or } (x', x) \text{ is a marked state of } S'_{k_1+k_2}] \wedge [\text{some cycle of } \text{Acc}(S) \text{ with nonempty label sequence is reachable from } x]\}$. Compute semiautomaton $S_{1+k_1+k_2}$ as follows: from each state of $X_{1+k_1+k_2}$, compute all transition sequences ended with a state $x$ that belongs to a cycle of $\text{Acc}(S)$ with nonempty label sequence, mark $x$. Regard $X_{1+k_1+k_2}$ as the initial state set of $S_{1+k_1+k_2}$ such that $S_{1+k_1+k_2}$ becomes an FSA. Remove every state $(x, x')$ of $S'_{k_1+k_2}$ such that neither $x$ nor $x'$ belongs to $X_{1+k_1+k_2}$.

6. in case of $k_1 > 1$ or $k_2 > 0$: Remove all states of $S'_{\min\{1, k_1\}} \cup \cdots \cup S'_{k_1+k_2-1}$ (and hence the corresponding transitions) from which none of states of $S'_{k_1+k_2}$ is reachable (in semiautomaton $S'_{\min\{1, k_1\}} \cup \cdots \cup S'_{k_1+k_2}$).

7. For $S_{1+k_1+k_2}$, try to choose to disable controllable transitions to cut off all transition sequences from initial states to marked states. If this can be done, then disabling these controllable transitions can make $S$ become $\omega$-($k_1, k_2$)-detectable. Otherwise, additionally choose other controllable transitions in $S'_{k_1+k_2}$, to disable to cut off transition sequences from $S'_{\min\{1, k_1\}}$ to $S'_{k_1+k_2}$ ended with a state $(x, x')$ of $S'_{k_1+k_2}$ such that $x$ (or $x'$) is an initial state of $S_{1+k_1+k_2}$ and a marked state is reachable from $x$ (or $x'$).

Let us intuitively explain the above procedure. Steps 1 through 4 computes semiautomata $S'_{\min\{1, k_1\}} \cup \cdots \cup S'_{k_1+k_2}$ with disjoint state sets, and collects all transition sequences in $\{5a\}$ (with possible redundant ones). Step 5 computes FSA $S_{1+k_1+k_2}$, and collects exactly all transition sequences in $\{5e\}$. Step 6 removes all redundant transition sequences obtained in Steps 1 through 4, which results in that semiautomaton $S'_{\min\{1, k_1\}} \cup \cdots \cup S'_{k_1+k_2}$ exactly collects all transition sequences in $\{5a\}$. Step 7 chooses controllable transitions to disable in order to cut off transition sequences from $S'_{\min\{1, k_1\}}$ to $S'_{k_1+k_2}$ and transitions from initial states to marked states in $S_{1+k_1+k_2}$ that violate $\omega$-($k_1, k_2$)-detectability.

**Example 4** Consider FSA $S$ shown in Fig. 8 and $\text{Acc}(\text{CC}_A(S))$ shown in Fig. 9. In $\text{Acc}(\text{CC}_A(S))$, there is a transition sequence $(s_0, (t_1, s_1)) \rightarrow (s_0, s_0) \rightarrow (s_1, s_1)$ with all events observable, where $(s_0, s_0)$ is an initial state, $(s_0, s_0) \rightarrow (s_0, s_0)$ and $(s_1, s_1) \rightarrow (s_1, s_1)$ are observable self-loops, and $s_1 \neq s_2$. In addition, in $\text{Acc}(S)$, there is a self-loop $s_1 \rightarrow s_1$ with $t_4$ observable. Then by the proof of Theorem 3, we have $S$ is not $\omega$-(1, 2)-detectable for any $k_1, k_2 \in \mathbb{N}$.
Fig. 8 An FSA.

Fig. 9 FSA Acc(CCₐ(S)), where FSA S is shown in Fig. 8.

We enforce $\omega$-(2, 2)-detectability of S. Following the above Steps 1 through 6, we draw as in Fig. 10. By Fig. 10, if transition $s₁ \xrightarrow{t₃} s₁$ is controllable, then after disabling this transition, in FSA $S_5$, there is no cycle with nonempty label sequence reachable from the unique initial state $s₁$ of $S_5$, and then FSA S becomes $\omega$-(2, 2)-detectable. Also by Fig. 10, if transition $s₀ \xrightarrow{t₁} s₁$ is controllable, then after disabling this transition, all transitions ended with a marked state of semiautomaton $S'_2$ will be cut off, and hence S also becomes $\omega$-(2, 2)-detectable.
Fig. 10 A figure used for enforcing $\omega$-$(2, 2)$-detectability of FSA $S$ shown in Fig. 8, where rectangle states denote the marked ones, dotted states are those that have been removed in Step 6 (i.e., state $(s_1, s_1)$ of semiautomaton $S'_4$ is not reachable from any of the dotted states).
5.2 Synthesizing \( *(k_1, k_2) \)-detectability

The procedure of synthesizing \( *(k_1, k_2) \)-detectability is quite similar to that of synthesizing \( \omega \)-(\( k_1, k_2 \))-detectability. We can also follow Steps 1 through 7 to synthesize \( *(k_1, k_2) \)-detectability, only if we remove Step 5, and also remove all modifications in Step 7 related to FSA \( S_{k_1+k_2} \). That is, Step 7 is changed to the one as follows:

- Choose controllable transitions in \( S'_{k_1+k_2} \) in order to disable to cut off transition sequences from \( S'_{\min(1,k_1)} \) to \( S'_{k_1+k_2} \).

See the following example.

**Example 5** Reconsider FSA \( S \) shown in Fig. 8 and \( \text{Acc}(CC_A(S)) \) shown in Fig. 9. In \( \text{Acc}(CC_A(S)) \), there is a transition sequence \((s_0,s_0) \xrightarrow{(t_1,t_1)} (s_0,s_0) \xrightarrow{(t_3,t_3)} (s_1,s_1) \xrightarrow{(t_4,t_4)} (s_1,s_1) \) with all events observable, where \((s_0,s_0) \) is an initial state, \((s_0,s_0) \xrightarrow{(t_1,t_1)} (s_0,s_0) \) and \((s_1,s_1) \xrightarrow{(t_4,t_4)} (s_1,s_1) \) are observable self-loops, and \( s_1 \neq s_2 \). Then by the proof of Theorem 5, we have \( S \) is not \( *(k_1, k_2) \)-detectable for any \( k_1, k_2 \in \mathbb{N} \).

We firstly enforce \( *(2,2) \)-detectability of \( S \). To this end, we only need to consider semiautomaton \( \bigcup_{i=1}^4 S'_i \) in Fig. 10. Similarly to Example 4, we do not need to consider dotted states and transitions. By Fig. 10, if transition \( s_1 \xrightarrow{t_4} s_4 \) is controllable, then after disabling this transition, the unique transition from \( S'_2 \) to \( S'_3 \) will be cut off, then \( S \) becomes \( *(2,2) \)-detectable. Similarly, if transition \( s_0 \xrightarrow{t_5} s_1 \) is controllable, then after disabling this transition, \( S \) also becomes \( *(2,2) \)-detectable.

Secondly, we enforce \( *(1,2) \)-detectability of \( S \). Following the above steps, we draw as in Fig. 11. By Fig. 11, if transition \( s_1 \xrightarrow{t_4} s_4 \) is controllable, then after disabling this transition, the unique transition from \( S'_2 \) to \( S'_3 \) will be cut off, then FSA \( S \) becomes \( *(1,2) \)-detectable. Similarly, if transition \( s_0 \xrightarrow{t_5} s_1 \) is controllable, then after disabling this transition, \( S \) also becomes \( *(1,2) \)-detectable. We also have if only transition \( s_0 \xrightarrow{t_5} s_0 \) is disabled, then \( S \) is still not \( *(1,2) \)-detectable.

Thirdly, we enforce \( *(0,2) \)-detectability of \( S \). Following the above steps, we draw the same picture as the case for enforcing \( *(1,2) \)-detectability as in Fig. 11 except that renaming \( S'_1, S'_2, S'_3 \) to \( S'_0, S'_1, S'_2 \). Then similar results could be obtained.

6 Concluding remarks

In this paper, we characterized a notion of \( K \)-delayed strong detectability for finite-state automata, and used a novel concurrent-composition method to give a quartic polynomial-time verification algorithm for the notion without any assumption. In addition, we obtained an upper bound for \( K \), and also studied two other similar notions of \( (k_1, k_2) \)-detectability and \((k_1, k_2)-D\)-detectability firstly studied in [13]. We also found quartic polynomial-time verification algorithms for the other two notions without any assumption, which strengthen the sextic polynomial-time verification algorithms given in [13] based on two widely-used assumptions. In addition, based on
Revisiting delayed strong detectability of discrete-event systems

Fig. 11 A figure used for enforcing $-\{1, 2\}$-detectability of FSA $S$ shown in Fig. 8, where rectangle states still denote the marked ones.

For our obtained results, we found polynomial-time algorithms for enforcing the above notions of delayed strong detectability, which are more effective than the exponential-time synthesis algorithms in the supervisory control framework in the literature.

Discussion on diagnosability

Next we briefly show that a slight variant of the concurrent composition can be used to obtain a polynomial-time verification algorithm for the notion of diagnosability of FSAs (originally studied in [10], exponential-time verification algorithms under Assumption 1 are given) without any assumption, which strengthens the polynomial-time verification algorithms given in [4,17] under several assumptions.

Consider an FSA $S = (X,T,X_0,\rightarrow,\Sigma,\ell)$, where additionally we partition the set $T$ of events into disjoint one subset $T_n$ of normal events and the other subset $T_f$ of faulty events. The notion of diagnosability studies whether one can make sure a faulty event has occurred after occurrences of a finite number of events. Next we state the notion of diagnosability.

**Definition 7** An FSA $S$ is called diagnosable with respect to $T_f$ if there is $K \in \mathbb{N}$, for every $s \in (T^*T_f) \cap T(S)$, for every $t \in T^*$ satisfying $st \in T(S)$ and $|t| \geq K$, $w \in T(S)$ and $\ell(w) = \ell(st)$ imply that $T_f \in w$ (which means that $w$ contains a faculty event).

By direction observation, one sees the following proposition on the notion of diagnosability.

**Proposition 2** An FSA $S$ is not diagnosable with respect to $T_f$ if and only if for all $K \in \mathbb{N}$, there is $s \in (T^*T_f) \cap T(S)$, $t \in T^*$, and $w \in T(S)$ such that $st \in T(S)$, $|t| \geq K$, $\ell(w) = \ell(st)$, and $T_f \notin w$ (which means that $w$ contains no faculty event).

Now consider a slight variant $CC_{A,n}^\prime(S)$ of concurrent composition $CC_A(S)$ of $S$ that is obtained from $CC_A(S)$ by changing $T_o$ to $\{(\bar{t},\bar{t}')|\bar{t} \in T, \bar{t}' \in T_n, \ell(\bar{t}) = \ell(\bar{t}')\}$.
and Definition 1. M. P. Cabasino, A. Giua, S. Lafortune, and C. Seatzu. A new approach for diagnosability analysis of Petri nets using verifier nets. *IEEE Transactions on Automatic Control*, 57(12):3104–3117, Dec 2012.

Theorem 8 An FSA $S = (X, T, X_0, \rightarrow, \Sigma, \ell)$ is not diagnosable with respect to $T_f$ if and only if there is $\\ell(\hat{t}) \in \Sigma$ and putting “with respect to a given $K$" under the linveness assumption given in [18] is of exponential-time complexity.

Definition 7 is exactly [1, Definition 5.2], where it is called *diagnosability in $K$ steps*. In [1], the focus is on labeled Petri nets, and another notion of diagnosability (the following Definition 8) has also been studied, and it is pointed out that these two notions are equivalent for FSAs [1, Proposition 5.3], although not equivalent for labeled Petri nets.

Definition 8 An FSA $S$ is called *diagnosable with respect to $T_f$* if for every $s \in (T^+T_f) \cap \mathcal{T}(S)$, there is $K_s \in \mathbb{N}$ such that, for every $t \in T^+$ satisfying $st \in \mathcal{T}(S)$ and $|t| \geqq K_s$, $w \in \mathcal{T}(S)$ and $\ell(w) = \ell(st)$ imply that $T_f \in w$.

By direction observation, one sees that an FSA $S$ is not diagnosable with respect to $T_f$ if and only if there is $s \in (T^+T_f) \cap \mathcal{T}(S)$, for all $K \in \mathbb{N}$, there exist $t \in T^+$ and $w \in \mathcal{T}(S)$ such that $st \in \mathcal{T}(S)$, $|t| \geqq K$, $\ell(w) = \ell(st)$, and $T_f \notin w$. This implies that negation of Definition 8 is also equivalent to satisfiability of (9), hence we obtain a different proof for the equivalence of Definition 8 and Definition 7.
2. C. G. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems*. Springer Publishing Company, Incorporated, 2nd edition, 2010.

3. C. N. Hadjicostis. *Estimation and Inference in Discrete Event Systems*. Communications and Control Engineering. Springer Nature Switzerland AG, 2020.

4. S. Jiang, Z. Huang, V. Chandra, and R. Kumar. A polynomial algorithm for testing diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control*, 46(8):1318–1321, Aug 2001.

5. C. Keroglou and C. N. Hadjicostis. Detectability in stochastic discrete event systems. *Systems & Control Letters*, 84:21–26, 2015.

6. T. Masopust. Complexity of deciding detectability in discrete event systems. *Automatica*, 93:257–261, 2018.

7. L. Mazare. Using unification for opacity properties. In *Proceedings of the Workshop on Issues in the Theory of Security (WITS’04)*, pages 165–176, 2004.

8. F. Pasqualetti, F. Dörfler, and F. Bullo. Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, 58(11):2715–2729, Nov 2013.

9. P. J. Ramadge and W. M. Wonham. Supervisory control of a class of discrete event processes. *SIAM Journal on Control and Optimization*, 25(1):206–230, 1987.

10. M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis. Diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control*, 40(9):1555–1575, Sep 1995.

11. S. Shu, Z. Huang, and F. Lin. Online sensor activation for detectability of discrete event systems. *IEEE Transactions on Automation Science and Engineering*, 10(2):457–461, April 2013.

12. S. Shu and F. Lin. Generalized detectability for discrete event systems. *Systems & Control Letters*, 60(5):310–317, 2011.

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

14. S. Shu, F. Lin, and H. Ying. Detectability of discrete event systems. *IEEE Transactions on Automatic Control*, 58(8):2125–2130, Aug 2013.

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

16. K. Zhang and A. Giua. K-delayed strong detectability of discrete-event systems. In *Proceedings of the 58th Conference on Decision and Control*, Nice, France, December 2019, accepted.