Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Instant detectability of discrete-event systems ⋆

Kuize Zhang * **Alessandro Giua** **

* *Control Systems Group at Technische Universität Berlin, Einsteinufer 17, 10587 Berlin, Germany (e-mail: zkz0017@163.com)*
** *Department of Electrical and Electronic Engineering, University of Cagliari, 09123 Cagliari, Italy, (e-mail: giua@unica.it)*

Abstract: Detectability is a basic property that describes whether an observer can use the *current* and *past* values of an observed output sequence produced by a system to reconstruct its *current* state. We consider particular properties called *instant strong detectability* and *instant weak detectability*, where the former implies that for each possible infinite observed output sequence each prefix of the output sequence allows reconstructing the current state, the latter implies that some infinite observed output sequence (if it exists) satisfies that each of its prefixes allows reconstructing the current state. For discrete-event systems modeled by finite-state automata, we give a linear-time verification algorithm for the former in the size of an automaton, and also give a polynomial-time verification algorithm for the latter.

Keywords: finite-state automaton, instant strong detectability, instant weak detectability

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 sequence produced by a system to reconstruct its *current* state [Shu et al. 2007, Shu and Lin 2011, 2013, Zhang 2017]. This property plays a fundamental role in many related control problems such as observer design and controller synthesis. Hence for different applications, it is meaningful to characterize different notions of detectability.

1.1 Literature review

For *discrete-event systems* (DESs) modeled by *finite-state automata*, the detectability problem has been widely studied [Shu et al. 2007, Shu and Lin 2011, Zhang 2017] in the context of the ω-*language* generated by a DES, i.e., taking into account all *infinite-length* (*infinite* for short) output sequences generated by the DES.

Two fundamental definitions are those of *strong detectability* and *weak detectability* [Shu et al. 2007]. Strong detectability implies that there exists a positive integer k such that for all infinite output sequences σ generated by a system, all prefixes of σ 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 σ generated by a system such that all prefixes of σ of length greater than k allow reconstructing the current states. Weak detectability is strictly weaker than strong detectability. Strong detectability can be verified in polynomial time (see a polynomial-time algorithm in [Shu and Lin 2011] under the widely-used *two fundamental assumptions* (formulated in Assumption 1) of *deadlock-freeness* and *promptness* (having no unobservable reachable cycle) and another polynomial-time algorithm given in [Zhang and Giua 2019] without any assumption), while weak detectability can be verified in exponential time [Shu et al. 2007]. In addition, checking weak detectability is PSPACE-complete [Zhang 2017, Masopust 2018].

A few authors have recently studied detectability properties in the context of ω-languages extending to labeled Petri nets the notions of strong and weak detectability which Shu and Lin have originally studied for finite-state automata. Weak detectability of labeled Petri nets with inhibitor arcs has been proved to be undecidable in [Zhang and Giua 2018]. Later the undecidable result has been strengthened to hold for labeled Petri nets in [Masopust and Yin 2019]. Strong detectability for labeled Petri nets has been proved to be decidable with EXPSPACE lower bound also in [Masopust and Yin 2019] under the two fundamental assumptions corresponding to labeled Petri nets. Moreover, the decidable result for strong detectability of labeled Petri nets has been strengthened to hold under only the promptness assumption corresponding to labeled Petri nets [Zhang and Giua 2020]. Detectability results on bounded labeled Petri nets can be found in [Tong et al. 2019].

1.2 Contribution of the paper

In many applications, e.g., those concerning safety-critical systems, it may be necessary to reconstruct the states at all times and thus neither strong detectability nor weak detectability meets the requirement. In order to meet the requirement, we consider instant detectability.

In this paper, we study *instant strong detectability* which implies that all prefixes of all infinite output sequences

⋆ This work was partially supported by the Alexander von Humboldt Foundation.

generated by a system allow reconstructing the current states. We also study *instant weak detectability* which implies that for some infinite output sequence generated by a system, each of its prefixes allows reconstructing the current state. In this paper, we only consider these two properties in the context of ω-languages, that is, we only consider long-term behavior. We point out that by using slight variants of the methods developed in the current paper, one can characterize their counterparts in the context of *formal languages* consisting of all generated *finite-length* (*finite* for short) output sequences. In more detail, in the context of formal languages, a system is instantly strongly (weakly) detectable if for every (some) finite output sequence generated by the system, each of its prefixes allows reconstructing the current state [1]. For instant strong detectability, the notion in the context of ω-languages is weaker than the one in the context of formal languages; while for instant weak detectability, the notion in the context of ω-languages does not imply the one in the context of formal languages, and vice versa. We also point out that although instant strong detectability in the context of ω-languages is weaker than its counterpart in the context of formal languages, the former implies that for every finite generated output sequence, most of its prefixes (except for at most a common number of longest prefixes, where the number is no less than the number of states) allow reconstructing the current states (formulated as an equivalent definition of instant strong detectability in the context of ω-languages, see Theorems 3.4 and 3.5), which is rather close to the latter. In the sequel, the two properties are always referred to as in the context of ω-languages by default.

The notion of instant strong detectability has been studied in [Shu and Lin 2013] for finite-state automata and is called $(0,0)$-detectability. Actually, a more general (k_1,k_2)-detectability is characterized in [Shu and Lin 2013] which describes a more general version of strong detectability with computation delays, and a sextic polynomial-time verification algorithm in the number of states is given under Assumption 1. More recently, a quartic polynomial-time algorithm in the number of states for verifying (k_1,k_2)-detectability of finite-state automata by using a concurrent-composition method has been given in [Zhang and Giua 2019] without any assumption, thus strengthening and simplifying the results given in [Shu and Lin 2013]. The notion of instant weak detectability is a new property.

The first contribution of this paper is for finite-state automata, we give a linear-time verification algorithm in the size of an automaton for instant strong detectability which exploits a particular characterization of this property and does not follow from the general but more complex approach for verifying (k_1,k_2)-detectability. In the particular characterization, we do not even use a concurrent-composition approach, hence the algorithm obtained in this paper is much more effective than the one designed in [Zhang and Giua 2019] for instant strong detectability.

[1] The notion of instant strong detectability in the context of formal languages for labeled Petri nets has been proved to be EXPSPACE-complete in [Jančar 1994], where the notion is called *determinism.*

The second contribution is that we design a polynomial-time verification algorithm for instant weak detectability of finite-state automata. This shows a remarkable difference between instant weak detectability and weak detectability, as weak detectability has PSPACE lower bound [Zhang 2017].

1.3 Paper structure

The remainder of the paper is as follows. Section 2 introduces necessary preliminaries. Section 3 shows the main results. Section 4 ends up with a short conclusion.

2. PRELIMINARIES

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 finite alphabet Σ, Σ^* and Σ^ω are used to denote the sets of finite sequences (called *words*) of elements of Σ including the empty word ϵ and infinite sequences (called *configurations*) of elements of Σ, respectively. As usual, we denote $\Sigma^+ = \Sigma^* \setminus \{\epsilon\}$. For a word $s \in \Sigma^*$, $|s|$ stands for its length, and we set $|s'| = +\infty$ for all $s' \in \Sigma^\omega$. For $s \in \Sigma$ and natural number k, s^k and s^ω denote the k-length word and configuration consisting of copies of s, respectively. For a word (configuration) $s \in \Sigma^*(\Sigma^\omega)$, a word $s' \in \Sigma^*$ is called a *prefix* of s, denoted as $s' \sqsubset s$, if there exists another word (configuration) $s'' \in \Sigma^*(\Sigma^\omega)$ such that $s = s's''$. For two natural numbers $i \le j$, $[i,j]$ 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.

2.1 Finite-state automata

A finite-state automaton is formulated as a sextuple

$$\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell),$$

where X is a finite set of states, T a finite set of events, $X_0 \subset X$ a set of initial states, $\to \subset X \times T \times X$ a transition relation, Σ a finite set of outputs, and $\ell : T \to \Sigma \cup \{\epsilon\}$ an output function. Automation $\mathcal{S}$ is called *deterministic* if for all $x, x', x'' \in X$ and $t \in T$, $(x,t,x') \in \to$ and $(x,t,x'') \in \to$ imply $x' = x''$. A state $x \in X$ is called *deadlock* if $(x,t,x') \notin \to$ for any $t \in T$ and $x' \in X$. $\mathcal{S}$ is called *deadlock-free* if it has no deadlock state. Events with label ϵ are called *unobservable.* Other events are called *observable.* Transitions $x \xrightarrow{t} x'$ with $\ell(t) = \epsilon$ are called ϵ-*transitions* (or *unobservable transitions*), other transitions are called *observable transitions.* Denote $T =: T_o \dot\cup T_\epsilon$, where T_o and T_ϵ are the sets of observable events and unobservable events, respectively. For an observable event $t \in T_o$, we say t *can be directly observed* if $\ell(t)$ differs from $\ell(t')$ for any other $t' \in T$. Labeling function $\ell : T \to \Sigma \cup \{\epsilon\}$ can be recursively extended to $\ell : T^* \cup T^\omega \to \Sigma^* \cup \Sigma^\omega$ as $\ell(t_1 t_2 \dots) = \ell(t_1)\ell(t_2)\dots$ and $\ell(\epsilon) = \epsilon$. Transition relation δ is recursively extended to $\delta \subset X \times T^* \times X$ as follows: for all $x, x' \in X$, $(x, \epsilon, x') \in \delta$ if and only if $x = x'$; for all $x, x' \in X$, $s \in T^*$, and $t \in T$, one has $(x, st, x') \in \delta$ (also denoted by $x \xrightarrow{st} x'$) if and only if $(x, s, x''), (x'', t, x') \in \delta$ for some $x'' \in X$. For two states $x, x' \in X$, we say x' *is reachable from* x if there is $s \in T^+$ such that $(x, s, x') \in \delta$; we say x' *is reachable* if either $x' \in X_0$ or x' is reachable from some initial state. More generally, for state $x \in X$ and subsets $X', X'' \subset X$, we say x *is reachable from* X' if x is reachable from some state of X'; X' *is reachable from* x if some state of X' is reachable from x; X'' *is reachable from* X' if some state of X'' is reachable from some state of X'. More generally, we call a *(transition) cycle* (i.e., $x \xrightarrow{s} x$ for some $s \in T^+$) *reachable* if some of its states is reachable (and hence all of its states are reachable). For a finite-state automaton $\mathcal{S}$, we call the new automaton the *accessible part* (denoted by $\mathrm{Acc}(\mathcal{S})$) of $\mathcal{S}$ that is obtained from $\mathcal{S}$ by removing all non-reachable states.

For each $\sigma \in \Sigma^*$, we denote by $\mathcal{M}(\mathcal{S}, \sigma)$ the set of states that $\mathcal{S}$ can be in after σ has been observed, i.e., $\mathcal{M}(\mathcal{S},\sigma) := \{x \in X | (\exists x_0 \in X_0)(\exists s \in T^*)[(\ell(s) = \sigma) \wedge (x_0 \xrightarrow{s} x)]\}$. Particularly, for all $X' \subset X$ and $\sigma \in \Sigma^*$ we denote $\mathcal{M}(X', \sigma) := \{x \in X | (\exists x' \in X')(\exists s \in T^*)[(\ell(s) = \sigma) \wedge (x' \xrightarrow{s} x)]\}$. Hence we have $\mathcal{M}(\mathcal{S}, \sigma) = \mathcal{M}(X_0, \sigma)$ and $X' \subset \mathcal{M}(X', \epsilon)$ for all $\sigma \in \Sigma^*$ and $X' \subset X$. $\mathcal{L}(\mathcal{S})$ denotes the *language generated* by system $\mathcal{S}$, i.e., $\mathcal{L}(\mathcal{S}) := \{\sigma \in \Sigma^* | \mathcal{M}(\mathcal{S},\sigma) \neq \emptyset\}$. An infinite event sequence $t_1 t_2 \ldots \in T^\omega$ is called *generated by* $\mathcal{S}$ if there exist states $x_0, x_1, \ldots \in X$ with $x_0 \in X_0$ such that for all $i \in \mathbb{N}$, $(x_i, t_{i+1}, x_{i+1}) \in \to$. We use $\mathcal{L}^\omega(\mathcal{S})$ to denote the *ω-language generated* by $\mathcal{S}$, i.e., $\mathcal{L}^\omega(\mathcal{S}) := \{\sigma \in \Sigma^\omega | (\exists t_1 t_2 \ldots \in T^\omega \text{ generated by } \mathcal{S})[\ell(t_1 t_2 \ldots) = \sigma]\}$.

The two assumptions widely considered in [Shu et al. 2007, Shu and Lin 2011, Sasi and Lin 2018] etc. are formulated as follows, but are not needed in the current paper.

Assumption 1: *A finite-state automaton $\mathcal{S}$ satisfies*

(i) $\mathcal{S}$ is deadlock-free, i.e., for each $x \in X$, there exist $t \in T$ and $x' \in X$ such that $(x, t, x') \in \to$;
(ii) $\mathcal{S}$ is prompt, i.e., there is no unobservable reachable cycle, i.e., for every reachable state $x \in X$, for every $s' \in (T_\epsilon)^+$, $(x, s', x) \notin \to$.

3. MAIN RESULTS

We now give the concepts of instant strong detectability and instant weak detectability.

Definition 1 (ISD)**:** *An automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$ is called instantly strongly detectable if for each σ in $\mathcal{L}^\omega(\mathcal{S})$, for each prefix σ' of σ, $|\mathcal{M}(\mathcal{S}, \sigma')| = 1$.*

Definition 2 (IWD)**:** *An automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$ is called instantly weakly detectable if $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$ implies that there exists $\sigma \in \mathcal{L}^\omega(\mathcal{S})$ such that each prefix σ' of σ satisfies $|\mathcal{M}(\mathcal{S}, \sigma')| = 1$.*

By definition, one sees instant weak detectability is weaker than instant strong detectability. In addition, if $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$, then $\mathcal{S}$ is naturally instantly strongly detectable, and hence also instantly weakly detectable.

3.1 Verifying instant strong detectability of finite-state automata

Consider a finite-state automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$. In order to verify instant strong detectability, we construct an *observation automaton*

$$\mathrm{O}(\mathcal{S}) = (X, \{\varepsilon, \hat{\epsilon}\}, X_0, \to', \{\hat{\epsilon}\}, \ell') \tag{1}$$

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

One sees by definition for automaton $\mathcal{S}$, in the corresponding observation automaton $\mathrm{O}(\mathcal{S})$, for every two states $x, x' \in X$, there is at most one transition from x to x' in $\mathrm{O}(\mathcal{S})$; there is an observable transition from x to x' in $\mathcal{S}$ if and only if $(x, \hat{\epsilon}, x') \in \to'$ (i.e., the unique transition from x to x' in $\mathrm{O}(\mathcal{S})$ is observable); there is an unobservable transition from x to x' and all transitions from x to x' are unobservable in $\mathcal{S}$ if and only if $(x, \varepsilon, x') \in \to'$ (i.e., the unique transition from x to x' in $\mathrm{O}(\mathcal{S})$ is unobservable).

Proposition 3.1: *For a finite-state automaton $\mathcal{S}$ it holds that $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$ if and only if $\mathcal{L}^\omega(\mathrm{O}(\mathcal{S})) \neq \emptyset$.*

Proof "if": Suppose $\mathcal{L}^\omega(\mathrm{O}(\mathcal{S})) \neq \emptyset$. Then there is an infinite transition sequence

$$x_0 \xrightarrow{\alpha_1} x_1 \xrightarrow{\alpha_2} \cdots \tag{2}$$

in $\mathrm{O}(\mathcal{S})$ such that $x_0 \in X_0$, $x_1, x_2, \ldots \in X$, $\alpha_1, \alpha_2, \ldots \in \{\varepsilon, \hat{\epsilon}\}$, and $\ell'(\alpha_1 \alpha_2 \ldots) = (\hat{\epsilon})^\omega$. For every transition $x_i \xrightarrow{\alpha_{i+1}} x_{i+1}$, where $i \in \mathbb{N}$, there is an observable transition from x_i to x_{i+1} in $\mathcal{S}$ if $\alpha_{i+1} = \hat{\epsilon}$, and there is an unobservable transition from x_i to x_{i+1} in $\mathcal{S}$ if $\alpha_{i+1} = \varepsilon$. Hence we can find an infinite transition sequence in $\mathcal{S}$ with the same state sequence as in (2), and the corresponding label sequence is of infinite length. Hence $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$.

The "only if" part holds similarly. □

The following useful proposition can be proved by computing strongly connected components of $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$.

Proposition 3.2: *The property $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$ for a finite-state automaton $\mathcal{S}$ can be verified in linear time in the size of $\mathcal{S}$.*

Proof Firstly, find the accessible part $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$ of $\mathrm{O}(\mathcal{S})$, which takes linear time. Secondly, by Proposition 3.1, we can equivalently check whether $\mathcal{L}^\omega(\mathrm{O}(\mathcal{S})) \neq \emptyset$ holds. Compute all strongly connected components of $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$. There are well-known algorithms for computing all strongly connected components of $\mathrm{Acc}(\mathcal{S})$ in linear time, e.g., the slight variant of the depth-first search. Thirdly, observe that $\mathcal{L}^\omega(\mathrm{O}(\mathcal{S})) \neq \emptyset$ if and only if in some strongly connected component, there is an observable transition, because each cycle belongs to only one strongly connected component. This can also be checked trivially in linear time. □

Consider a finite-state automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$. In order to verify instant strong detectability, we also need to construct a *bifurcation automaton*

$$\mathrm{Bifur}(\mathcal{S}) = (X, \{\bar{\epsilon}, \check{\epsilon}\}, X_0, \to', \{\bar{\epsilon}, \check{\epsilon}\}, \ell') \tag{3}$$

in linear time of the size of $\mathcal{S}$, where $\to' \subset X \times \{\bar{\epsilon}, \check{\epsilon}\} \times X$, $\ell'(\bar{\epsilon}) = \bar{\epsilon}$, $\ell'(\check{\epsilon}) = \check{\epsilon}$, ℓ' is also naturally extended to $\ell' : \{\bar{\epsilon}, \check{\epsilon}\}^* \cup \{\bar{\epsilon}, \check{\epsilon}\}^\omega \to \{\bar{\epsilon}, \check{\epsilon}\}^* \cup \{\bar{\epsilon}, \check{\epsilon}\}^\omega$, transitions $x \xrightarrow{\bar{\epsilon}} x'$ are called *fair transitions*, transitions $x \xrightarrow{\check{\epsilon}} x'$ are called *bifurcation transitions*; for every two states $i, j \in X$, (i) $(j, \bar{\epsilon}, i), (j, \check{\epsilon}, i) \notin \to'$ if $\neg A_1$, (ii) $(x, \bar{\epsilon}, x') \in \to'$ if $A_1 \wedge A_2 \wedge A_3$, (iii) $(x, \check{\epsilon}, x') \in \to'$ otherwise, where

$$A_1 = (\exists t \in T)[(j, t, i) \in \to],$$

$$
\begin{aligned}
A_2 =& (\nexists t \in T, j' \in X) \\
& [((j,t,j') \in \to) \wedge (\ell(t)=\epsilon) \wedge (j' \neq j)], \\
A_3 =& (\forall t \in T)[(((j,t,i) \in \to) \wedge (\ell(t) \neq \epsilon)) \implies \\
& (\nexists t' \in T, j' \in X) \\
& [((j,t',j') \in \to) \wedge (\ell(t') = \ell(t)) \wedge (j' \neq i)]].
\end{aligned}
$$

One sees that both fair transitions and bifurcations transitions can be unobservable transitions or observable transitions. Next we explain the relation between $\mathrm{Bifur}(\mathcal{S})$ and the original automaton $\mathcal{S}$. Here (i) holds if there is no transition from state j to state i in $\mathcal{S}$; (ii) holds if there exists a transition from j to i, and none of such transitions has a bifurcation in $\mathcal{S}$; and (iii) holds if there is a transition from j to i that has a bifurcation also in $\mathcal{S}$. For the case that (iii) holds, if A_1 holds but A_2 does not hold, then for $\mathcal{S}$ one has $\{j\} \subsetneq \mathcal{M}(\{j\},\epsilon)$ and hence $|\mathcal{M}(\{j\},\epsilon)| > 1$; if A_1 and A_2 hold but A_3 does not hold, then for $\mathcal{S}$ one has $|\mathcal{M}(\{j\},\epsilon)| = 1$, $\{i\} \subsetneq \mathcal{M}(\{j\},\ell(\tilde{t}'))$, and hence $|\mathcal{M}(\{j\},\ell(\tilde{t}'))| > 1$ for some $\tilde{t}' \in T$ with $\ell(\tilde{t}') \neq \epsilon$ and $(j,\tilde{t}',i) \in \to$.

Simlarly to $\mathrm{O}(\mathcal{S})$, in $\mathrm{Bifur}(\mathcal{S})$, one also has for every two states $x, x' \in X$, there is at most one transition from x to x'. If in $\mathcal{S}$, there is a transition sequence $x_0 \xrightarrow{s} x \xrightarrow{t} x'$ with $x_0 \in X_0$ such that $x \xrightarrow{\check{\epsilon}} x'$ is a bifurcation transition of $\mathrm{Bifur}(\mathcal{S})$, then either $|\mathcal{M}(\mathcal{S},\ell(s))| > 1$ or $\mathcal{M}(\mathcal{S},\ell(st'))| > 1$ for some $t' \in T_o$ such that $(x,t',x') \in \to$. Hence the occurrence of a bifurcation transition makes an observed output sequence not allow reconstructing the current state. One also has that for all states x and x', there is a transition from x to x' in $\mathcal{S}$ if and only if there is a transition from x to x' in $\mathrm{O}(\mathcal{S})$ if and only if there is a transition from x to x' in $\mathrm{Bifur}(\mathcal{S})$. This obvious observation is helpful in verifying instant strong detectability.

Theorem 3.3: *The instant strong detectability of finite-state automata can be verified in linear time.*

Proof Consider a finite-state automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$ and its bifurcation automaton $\mathrm{Bifur}(\mathcal{S})$ defined by (3). If $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$, then $\mathcal{S}$ is naturally instantly strongly detectable. By Proposition 3.2, it takes linear time of the size of $\mathcal{S}$ to check whether $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$. Next we assume that $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$. If additionally $|X_0| > 1$, then by definition $\mathcal{S}$ is not instantly strongly detectable either. Next we additionally assume that there is a unique initial state.

We claim that $\mathcal{S}$ is not instantly strongly detectable if and only if in $\mathcal{S}$, there is a transition sequence

$$x_0 \xrightarrow{s_1} x_1 \xrightarrow{t} x_2 \xrightarrow{s_2} x_3 \xrightarrow{s_3} x_3 \tag{4}$$

with $x_0 \in X_0$, $x_1, x_2, x_3 \in X$, $s_1, s_2, s_3 \in T^*$, $t \in T$ such that $\ell(s_3) \in \Sigma^+$ and there is a bifurcation transition $x_1 \xrightarrow{\check{\epsilon}} x_2$ in $\mathrm{Acc}(\mathrm{Bifur}(\mathcal{S}))$.

"if": This holds because the cycle $x_3 \xrightarrow{s_3} x_3$ with positive-length label sequence can be extended to an infinite-length transition sequence with infinite-length label sequence, and either $|\mathcal{M}(\{x_0\},\ell(s_1))| > 1$ or $|\mathcal{M}(\{x_0\},\ell(s_1 t'))| > 1$ for some $t' \in T$ such that $\ell(t') \neq \epsilon$ and $(x_1, t', x_2) \in \to$ by the notion of bifurcation automaton.

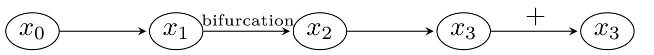

Fig. 1. A sketch for verifying instant strong detectability of finite-state automata.

"only if": If $\mathcal{S}$ is not instantly strongly detectable, then there is an infinite transition sequence $x_0 \xrightarrow{s_1} \bar{x} \xrightarrow{s_2}$ and a finite transition sequence $x_0 \xrightarrow{s_1'} \bar{x}'$ such that $x_0 \in X_0$, $\bar{x}, \bar{x}' \in X$, $\bar{x} \neq \bar{x}'$, $s_1, s_1' \in T^*$, $\ell(s_1) = \ell(s_1')$, $s_2 \in T^\omega$, and $\ell(s_2) \in \Sigma^\omega$. Then $s_1, s_1' \in T^+$ since at least one of $\bar{x}$ and $\bar{x}'$ differs from x_0. Moreover, $|\mathcal{M}(\{x_0\},\ell(s_1))| > 1$. Hence in the transition sequence of $\mathrm{Acc}(\mathrm{Bifur}(\mathcal{S}))$ that has the same state sequence as $x_0 \xrightarrow{s_1} \bar{x}$ of $\mathcal{S}$, there is a bifurcation transition. In addition, by the finiteness of X and $\ell(s_2) \in \Sigma^\omega$, in $\mathcal{S}$ there is a cycle with positive-length label sequence reachable from $\bar{x}$.

We next check the above equivalent condition for instant strong detectability. See Fig. 1 for a sketch.

(1) Compute the accessible part $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$ of the observation automaton $\mathrm{O}(\mathcal{S})$ of $\mathcal{S}$ defined by (1).
(2) Compute the set X_c of states x_3 of $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$ that belong to a cycle of $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$ with positive-length label sequence. (Then we have $X_c \neq \emptyset$ by $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$.)
(3) Compute $\mathrm{Acc}(\mathrm{Bifur}(\mathcal{S}))$.
(4) Check whether there is a bifurcation transition $x_1 \xrightarrow{\check{\epsilon}} x_2$ in $\mathrm{Acc}(\mathrm{Bifur}(\mathcal{S}))$ such that X_c is reachable from x_2. Such a bifurcation transition $x_1 \xrightarrow{\check{\epsilon}} x_2$ exists if and only if transition sequence (4) exists.

The first step and the third step both take linear time.

For the second step, we firstly compute all strongly connected components of $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$ in linear time. Observe that for each strongly connected component, if it contains a transition, then it contains a cycle containing all its states and transitions. One then has that the set X_c consists of all states of all strongly connected components of $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$ containing at least one observable transition. Hence X_c can be computed in linear time.

Recall that $\mathrm{Acc}(\mathrm{Bifur}(\mathcal{S}))$ and $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$ have the same set of states, and for every two states x and x', there is a transition from x to x' in $\mathrm{Acc}(\mathrm{Bifur}(\mathcal{S}))$ if and only if there is a transition also from x to x' in $\mathrm{Acc}(\mathrm{O}(\mathcal{S}))$. Then the fourth step consumes linear time of $\mathcal{S}$ by traversing from X_c all paths along the inverse direction of transitions. $\square$

Example 1: *Consider the finite-state automaton $\mathcal{S}$ in the left part of Fig. 2. Its observation automaton and bifurcation automaton are shown in the middle part and the right part of Fig. 2, respectively. It has a unique initial state and generates a nonempty ω-language. In addition, all its states are reachable. According to the proof of Theorem 3.3, one then has $X_c = \{s_0, s_1\}$, and in its bifurcation automaton there is a transition $s_0 \xrightarrow{\check{\epsilon}} s_1$ such that s_1 in X_c is reachable from s_1 in the transition. Then $\mathcal{S}$ is not instantly strongly detectable.*

Next we give an equivalent representation for instant strong detectability by using formal languages, which shows a particular interest of the property. That is, for automaton $\mathcal{S}$ that generates a nonempty ω-language, it

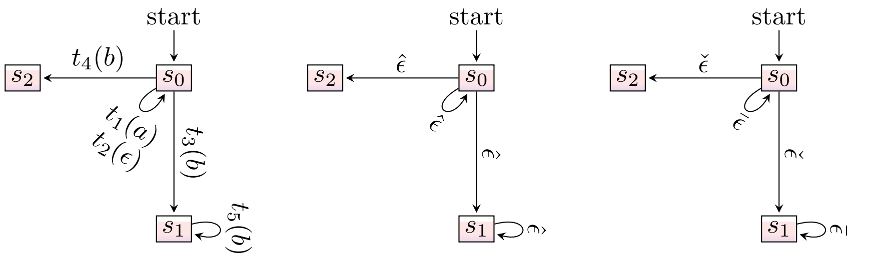

Fig. 2. A finite-state automaton (left), its observation automaton (middle), and its bifurcation automaton (right).

is instantly strongly detectable if and only if for every $\sigma \in \mathcal{L}(\mathcal{S})$, all but at most a number $|X|-1$ of longest ones of its prefixes allow reconstructing the current states.

Theorem 3.4: *Consider a finite-state automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$ such that $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$. It is instantly strongly detectable if and only if every $\sigma \in \mathcal{L}(\mathcal{S})$ can be written as $\sigma = \sigma_1\sigma_2$, where $|\mathcal{M}(\mathcal{S}, \sigma_1')| = 1$ for all $\sigma_1' \sqsubset \sigma_1$ and $|\sigma_2| < |X|$.*

Proof "if": For all $\bar{\sigma} \in \mathcal{L}^\omega(\mathcal{S})$ and all $\bar{\sigma}_1 \sqsubset \bar{\sigma}$, choose $\bar{\sigma}_2$ such that $\bar{\sigma}_1\bar{\sigma}_2 \sqsubset \bar{\sigma}$ and $|\bar{\sigma}_2| = |X|$, then $\bar{\sigma}_1\bar{\sigma}_2 \in \mathcal{L}(\mathcal{S})$, and $|\mathcal{M}(\mathcal{S}, \bar{\sigma}_1)| = 1$, implying $\mathcal{S}$ is instantly strongly detectable.

"only if": If $\sigma \in \mathcal{L}(\mathcal{S})$ is a prefix of some configuration of $\mathcal{L}^\omega(\mathcal{S})$, then one has $|\mathcal{M}(\mathcal{S}, \sigma')| = 1$ for all $\sigma' \sqsubset \sigma$. Next we suppose that σ is not a prefix of any configuration of $\mathcal{L}^\omega(\mathcal{S})$. If $|\sigma| < |X|$, then we choose $\sigma_1 = \epsilon$ and $\sigma_2 = \sigma$, and then we have $|\mathcal{M}(\mathcal{S}, \sigma_1)| = 1$, otherwise $\mathcal{S}$ is not instantly strongly detectable. Next we also suppose $|\sigma| \geq |X|$. Write $\sigma = \alpha_1 \dots \alpha_{|\sigma|}$ with each α_i in Σ. Choose an arbitrary transition sequence $x_0 \xrightarrow{s_1} \cdots \xrightarrow{s_{|\sigma|}} x_{|\sigma|}$ such that $x_0 \in X_0$, $x_1, \dots, x_{|\sigma|} \in X$, $s_i \in T^+$, $\ell(s_i) = \alpha_i$, for all $i \in [1, |\sigma|]$. Then $x_{|\sigma|}$ differs from any of the other states $x_0, \dots, x_{|\sigma|-1}$, otherwise σ is a prefix of some configuration of $\mathcal{L}^\omega(\mathcal{S})$. By the Pigeonhole Principle, there exist $|\sigma| - |X| \leq k < l \leq |\sigma| - 1$ such that $x_k = x_l$. Hence $\sigma_1 \dots \sigma_k(\sigma_{k+1} \dots \sigma_l)^\omega \in \mathcal{L}^\omega(\mathcal{S})$, and $|\mathcal{M}(\mathcal{S}, \sigma_1 \dots \sigma_i)| = 1$ for all $i \in [1, l]$ and $|\mathcal{M}(\mathcal{S}, \epsilon)| = 1$, since $\mathcal{S}$ is instantly strongly detectable. One also has $|\sigma| - l < |X|$, which completes the proof. □

Similarly to Theorem 3.4, the following a little bit weaker equivalent condition for instant strong detectability holds without requiring automaton $\mathcal{S}$ to generate a nonempty ω-language. That is, $\mathcal{S}$ is instantly strongly detectable if and only if for every $\sigma \in \mathcal{L}(\mathcal{S})$ of length no less than $|X|$, all but at most a number $|X|-1$ of longest ones of its prefixes allow reconstructing the current states. We omit a similar proof.

Theorem 3.5: *A finite-state automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$ is instantly strongly detectable if and only if every $\sigma \in \mathcal{L}(\mathcal{S})$ satisfying $|\sigma| \geq |X|$ can be written as $\sigma = \sigma_1\sigma_2$, where $|\mathcal{M}(\mathcal{S}, \sigma_1')| = 1$ for all $\sigma_1' \sqsubset \sigma_1$ and $|\sigma_2| < |X|$.*

3.2 Verifying instant weak detectability of finite-state automata

A notion of *observer* has been used to verify weak detectability [Shu et al. 2007]. Due to the similarity between weak detectability and instant weak detectability, we firstly use the observer to give an exponential-time verification algorithm for instant weak detectability, and then obtain a polynomial-time algorithm by reducing the observer.

Consider automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$. The *unobservable reach* of X_0 is defined by $\mathrm{UR}(X_0) := \{x' \in X | (\exists x_0 \in X_0)(\exists s \in (T_\epsilon)^*)[(x_0, s, x') \in \to]\}$. For a subset $X' \subset X$, its *observable reach under* $\sigma \in \Sigma$ is defined by $\mathrm{Reach}_\sigma(X') := \{x' \in X | (\exists x \in X')(\exists t \in T_o)(\exists s \in (T_\epsilon)^*)[(\ell(t) = \sigma) \wedge ((x, ts, x') \in \to)]\}$.

The observer of $\mathcal{S}$ is defined by a deterministic automaton

$$\mathcal{S}_{obs} = (Q, \Sigma, \{q_0\}, \to_{obs}),$$

where $Q \subset 2^X \setminus \{\emptyset\}$ is the state set, $q_0 = \mathrm{UR}(X_0)$ is the unique initial state, $\to_{obs} \subset Q \times \Sigma \times Q$ is the transition relation and also extended recursively to $\to_{obs} \subset Q \times \Sigma^* \times Q$, for every transition sequence $(q_0, \sigma, q) \in \to_{obs}$ with $\sigma \in \Sigma^+$, $q = \mathcal{M}(\mathcal{S}, \sigma)$.

Theorem 3.6: *Automaton $\mathcal{S}$ is instantly weakly detectable if and only if either $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$ or in its observer $\mathcal{S}_{obs}$, there is a transition sequence from q_0 to a cycle such that all states in the sequence and the cycle, including q_0, are singletons.*

Proof This result follows from the finiteness of the number of states and the Pigeonhole Principle. □

The verification algorithm derived from Theorem 3.6 runs in exponential time. Next, we improve this result to obtain a polynomial-time verification algorithm. Consider automaton $\mathcal{S}$ and its observer $\mathcal{S}_{obs}$, construct a new automaton

$$\mathcal{S}_{obs}^s = (Q, \Sigma, \{q_0\}, \to_{obs}^s),$$

where $\to_{obs}^s \subset \to_{obs}$, for every transition $(q_1, \sigma, q_2) \in \to_{obs}$, $(q_1, \sigma, q_2) \in \to_{obs}^s$ if and only if q_1, q_2 are singletons. Then the following result directly follows from Theorem 3.6.

Theorem 3.7: *Automaton $\mathcal{S}$ is instantly weakly detectable if and only if either $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$ or in automaton $\mathrm{Acc}(\mathcal{S}_{obs}^s)$, there is a cycle.*

Automaton $\mathrm{Acc}(\mathcal{S}_{obs}^s)$ can be computed in polynomial time in the size of $\mathcal{S}$. Furthermore, we obtain the polynomial-time Algorithm 1 for verifying instant weak detectability.

Now we show that Algorithm 1 returns the correct answer in polynomial time. In Line 1, by Proposition 3.2, it takes linear time in the size of $\mathcal{S}$ to check whether $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$ holds. By definition, if $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$ then $\mathcal{S}$ is instantly weakly detectable. Next we assume that $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$. In Line 9, it takes time $|X|^2|T_\epsilon|$ to compute $\mathrm{UR}(X_0)$. By Theorem 3.7, if $|\mathrm{UR}(X_0)| > 1$, then $\mathcal{S}$ is not instantly weakly detectable. Next we assume $|\mathrm{UR}(X_0)| = 1$. For

Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Algorithm 1

Input: A finite-state automaton $\mathcal{S} = (X, T, X_0, \to, \Sigma, \ell)$
Output: "YES" if $\mathcal{S}$ is instantly weakly detectable and "NO" otherwise

1: **if** $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$ **then**
2:   **return** YES
3:   **break**
4: **else**
5:   **if** $|X_0| > 1$ **then**
6:     **return** NO
7:     **break**
8:   **else**
9:     $q_0 := \mathrm{UR}(X_0)$
10:     **if** $|q_0| > 1$ **then**
11:       **return** NO
12:       **break**
13:     **else**
14:       Initiate automaton $\mathrm{Acc}(\mathcal{S}^s_{obs}) := (Q', \Sigma, \{q_0\}, \to^s_{obs})$, where $Q' = \{q_0\}$, $\to^s_{obs} = \emptyset$
15:       $Q_1 := \emptyset$, $Q_2 := \{q_0\}$
16:       **while** $Q_2 \neq \emptyset$ **do**
17:         **for** $q \in Q_2$ **do**
18:           **for** $\sigma \in \Sigma$ **do**
19:             **if** $|\,\mathrm{Reach}_\sigma(q)| = 1$ **then**
20:               $\to^s_{obs} := \to^s_{obs} \cup \{(q, \sigma, \mathrm{Reach}_\sigma(q))\}$
21:               **if** $\mathrm{Reach}_\sigma(q) \notin Q'$ **then**
22:                 $Q_1 := Q_1 \cup \{\mathrm{Reach}_\sigma(q)\}$
23:               **end if**
24:             **end if**
25:           **end for**
26:         **end for**
27:         $Q' := Q' \cup Q_1$, $Q_2 := Q_1$, $Q_1 := \emptyset$
28:       **end while**
29:       **if** there is a cycle in $\mathrm{Acc}(\mathcal{S}^s_{obs})$ **then**
30:         **return** YES
31:         **break**
32:       **else**
33:         **return** NO
34:         **break**
35:       **end if**
36:     **end if**
37:   **end if**
38: **end if**

every $x \in X$ and every $\sigma \in \Sigma$, it takes time $|T_o||X| + |X|^2|T_\epsilon|$ to compute $\mathrm{Reach}_\sigma(\{x\})$. Hence it takes time $|T_o||X|^2 + |X|^3|T_\epsilon|$ to finish the **while** structure (Lines 16 through 28). After the **while** structure, automaton $\mathrm{Acc}(\mathcal{S}^s_{obs})$ in Theorem 3.7 is generated. In Line 29, it takes linear time in the size of the final $\mathrm{Acc}(\mathcal{S}^s_{obs})$ to check whether there is a cycle in $\mathrm{Acc}(\mathcal{S}^s_{obs})$. Again by Theorem 3.7, $\mathcal{S}$ is instantly weakly detectable if there is a cycle, and not instantly weakly detectable otherwise.

Example 2: *Reconsider the automaton $\mathcal{S}$ shown in the left part of Fig. 2. Its observer $\mathcal{S}_{obs}$ is shown in the left part of Fig. 3, and the corresponding $\mathrm{Acc}(\mathcal{S}^s_{obs})$ is shown in the right part of Fig. 3. By Theorem 3.6 and $\mathcal{S}_{obs}$, one has $\mathcal{S}$ is instantly weakly detectable, as $a^\omega \in \mathcal{L}^\omega(\mathcal{S})$, and $\mathcal{M}(\mathcal{S}, a^n) = \{s_0\}$ for all $n \in \mathbb{N}$. On the other hand, by Algorithm 1, since in $\mathrm{Acc}(\mathcal{S}^s_{obs})$ there is a cycle, then one also has $\mathcal{S}$ is instantly weakly detectable.*

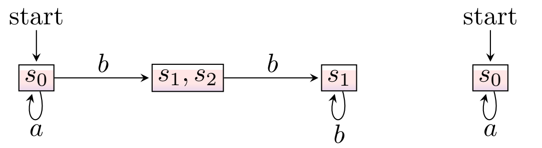

Fig. 3. Observer $\mathcal{S}_{obs}$ of the automaton $\mathcal{S}$ in the left part of Fig. 2 (left), and the corresponding automaton $\mathrm{Acc}(\mathcal{S}^s_{obs})$ (right).

4. CONCLUSION

In this paper, we studied results on instant strong detectability and instant weak detectability of finite-state automata. We gave fast polynomial-time verification algorithms for both properties. How to extend these results to labeled Petri nets are future interesting questions.

REFERENCES

Jančar, P. (1994). Decidability questions for bisimilarity of Petri nets and some related problems. In P. Enjalbert, E.W. Mayr, and K.W. Wagner (eds.), *STACS 94*, 581–592. Springer Berlin Heidelberg, Berlin, Heidelberg.

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

Masopust, T. and Yin, X. (2019). Deciding detectability for labeled Petri nets. *Automatica*, 104, 238–241.

Sasi, Y. and Lin, F. (2018). Detectability of networked discrete event systems. *Discrete Event Dynamic Systems*, 28(3), 449–470.

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

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

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

Tong, Y., Lan, H., and Guo, J. (2019). Verification of detectability in labeled Petri nets. In *2019 American Control Conference (ACC)*, 5627–5632.

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

Zhang, K. and Giua, A. (2018). Weak (approximate) detectability of labeled Petri net systems with inhibitor arcs. *IFAC-PapersOnLine*, 51(7), 167–171.

Zhang, K. and Giua, A. (2019). K-delayed strong detectability of discrete-event systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, 7647–7652.

Zhang, K. and Giua, A. (2020). On detectability of labeled Petri nets and finite automata. *Discrete Event Dynamic Systems*, in press, 33 pages.