

On detectability of labeled Petri nets and finite automata*

Kuize Zhang

Alessandro Giua

Abstract

Detectability is a basic property of dynamic systems: when it holds an observer can use the *current* and *past* values of the observed output signal produced by a system to reconstruct its *current* state.

In this paper, we consider properties of this type in the framework of discrete-event systems modeled by labeled Petri nets and finite automata. We first study *weak approximate detectability*. This property implies that there exists an infinite observed output sequence of the system such that each prefix of the output sequence with length greater than a given value allows an observer to determine if the current state belongs to a given set. We prove that the problem of verifying this property is undecidable for labeled Petri nets, and PSPACE-complete for finite automata.

We also consider one new concept called *eventual strong detectability*. The new property implies that for each possible infinite observed output sequence, there exists a value such that each prefix of the output sequence with length greater than that value allows reconstructing the current state. We prove that for labeled Petri nets, the problem of verifying eventual strong detectability is decidable and EXPSPACE-hard, where the decidability result holds under a mild promptness assumption. For finite automata, we give a polynomial-time verification algorithm for the property. In addition, we prove that strong detectability is strictly stronger than eventual strong detectability for labeled Petri nets and even for deterministic finite automata.

Published as:

K. Zhang, A. Giua, "On detectability of labeled Petri nets and finite automata," *Discrete Event Dynamic Systems*, Vol. 30, pp. 465–497, 2020. DOI: 10.1007/s10626-020-00311-3

*K. Zhang is with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, 10044 Stockholm, Sweden, A. Giua is with the Department of Electrical and Electronic Engineering, University of Cagliari, 09123 Cagliari, Italy.

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 signal produced by a system to reconstruct its *current* state [9, 30, 27, 28, 8, 36, 48, 23, 41, 16, 12, 25]. 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. This property also has different terminologies, e.g., in [9, 36, 23], it is called “observability” while in [8, 48], it is called “reconstructibility”. In this paper, we uniformly call this property “detectability”, and call another similar property “observability” implying that the *initial state* can be determined by the observed output signal produced by a system (e.g., [38, 29, 46, 47]).

1.1 Literature review

Finite automata

For *discrete-event systems* (DESs) modeled by *finite automata*, the detectability problem has been widely studied [30, 27, 44, 16, 41] in the context of ω -*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 formally stated in the following sections: when it holds, a system will always run and generate an infinitely long observation.

Two fundamental definitions are those of *strong detectability* and *weak detectability* [30]. Strong detectability implies¹ that:

(A) 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:

(B) 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. Consider the finite automaton shown in Fig. 1, where events a and b can be directly observed. It is weakly detectable but not strongly detectable. The automaton can generate infinite event sequences a^ω and b^ω , where $(\cdot)^\omega$ denotes

¹Formal definitions of strong and weak detectability are given later in Definitions 1 and 3.

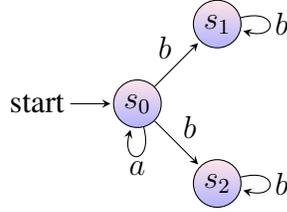


Figure 1: A finite automaton.

the concatenation of infinitely many copies of \cdot . When any number of a 's were observed but no b was observed, the automaton could be only in state s_0 . Hence it is weakly detectable. When any number of b 's were observed but no a was observed, it could be in states s_1 or s_2 . Hence it is not strongly detectable.

Strong detectability can be verified in polynomial time while weak detectability can be verified in exponential time [30, 27] under the usual Assumption 1.

In addition, checking weak detectability is PSPACE-complete in the numbers of states and events for finite automata, where the hardness result holds for deterministic finite automata whose events can be directly observed [44]. The hardness result even holds for more restricted deterministic finite automata having only two events that can be directly observed [16].

Petri nets

Detectability of free-labeled Petri nets with unknown initial *markings* (i.e., states) has been studied in [9], where several types of detectability called “(strong) marking observability”, “uniform (strong) marking observability”, and “structural (strong) marking observability” are proved to be decidable² by reducing them to several decidable home space properties [6] that are more general than the reachability problem of Petri nets (with respect to a given marking).

Some detectability properties of *labeled Petri nets*³ have also been studied. In [23], a notion of detectability called “structural observability” is characterized. This property implies that for every initial marking, each observed *label* (i.e., output) sequence determines the current marking. It is pointed out that the “structural observability” is important, because “the majority of existing control schemes for Petri nets rely on complete knowledge of the system state at any given time step” [23]. It is shown that structural observability can be verified in polynomial time [23]. In the same paper, in order to make a labeled Petri net structurally observable, the problem of placing the minimal number of sensors on places and the problem of placing the minimal number of sensors on transitions are studied, respectively. The former problem is proved to be NP-complete, while the latter is shown to be solvable in polynomial time, both in the numbers of places and transitions.

In [11], for labeled Petri nets, a concept of determinism is characterized, where this concept implies

²In the sequel, we will always use the expression “a property is decidable/undecidable” instead of “the problem of verifying the property is decidable/undecidable.”

³More precisely *labeled place/transition nets* or *labeled P/T nets* for short

that each label sequence generated by a net can be used to determine the current marking. It is proved that verifying determinism is as hard as verifying *coverability* for Petri nets [21, 15], hence EXPSPACE-complete. Note that the “structural observability” studied in [23] requires a labeled Petri net to satisfy the determinism property at each initial marking.

The above mentioned detectability results for labeled Petri nets apply to finite-length languages of the nets, i.e., the set of all words (of finite length) that a net can generate. In the sequel, we always use terminology “language” to denote “finite-length language” for short, and use “ ω -language” to denote a “language” consisting of several infinite-length label sequences. However, a few authors have recently studied detectability properties of ω -languages extending to labeled Petri net models the notions of strong and weak detectability which Shu and Lin have originally studied in the context of finite automata.

Weak detectability of labeled Petri nets with inhibitor arcs has been proved to be undecidable in [45] by reducing the well known undecidable *language equivalence problem* [10, Theorem 8.2] of labeled Petri nets to the inverse problem of the weak detectability problem, i.e., the non-weak detectability problem.

Decidability and complexity of strong detectability and weak detectability for labeled Petri nets are also studied in [17]. Under [the first item](#) of Assumption 1 and another assumption that a net cannot generate an infinite unobservable sequence which is actually equivalent to [the second item](#) of Assumption 1 for Petri nets, strong detectability has been proved to be decidable with EXPSPACE-hard complexity in [17] by reducing its negation to the satisfiability of a *Yen’s path formula* ([37, 2]). Weak detectability has been proved to be undecidable by reducing the undecidable *language inclusion problem* [10, Theorem 8.2] to the non-weak detectability problem, thus improving the related result given in [45].

1.2 Contribution of the paper

In this paper, we propose some new notions of detectability in the context of ω -languages, and characterize the related decision problems (in terms of decidability or computational complexity) for both finite automata and labeled Petri nets.

To motivate the interest for this work, let us recall that the theory of ω -languages is a rich and important domain of computer science [20]. We mention, in addition, that these languages have a practical interest in automatic control because they can describe the infinite behavior of a system: for this reason they find significant applications in the very active area of verification with discrete-event and hybrid systems — in particular model checking with temporal logic.

Eventual detectability

Let us consider again the notion of strong detectability implied by condition (A) stated above. An alternative definition could be based on the following definition:

(A') for every infinite output sequence σ generated by a system, there exists a positive integer k_σ such that all prefixes of σ of length greater than k_σ allow reconstructing the current states,

where the length k_σ of the transient before the state can be reconstructed may depend on a particular output sequence σ .

Obviously, condition (A) implies condition (A') but the converse implication does not hold, because there may exist infinitely many strings of infinite length and thus a maximal value among all k_σ may not be computed (this will be formally proved in Proposition 4).

We point out some similarities with the notion of diagnosability introduced by Lafortune and co-authors [24] which requires the occurrence of a fault to be detected within a finite delay. The original definition in [24] assumes this delay may depend on the string that produces the fault, i.e., it is similar to condition (A') above. A different condition, similar to condition (A) above and called *K-step diagnosability* or *uniform diagnosability*, is considered in [3] or [43]: it assumes the length of the delay is bounded for all strings. Note however a difference with respect to the detectability results we present here: the two notions of diagnosability and *K-step diagnosability* are equivalent in the case of finite automata, thanks to the well-known Myhill-Nerode characterization of a regular language by the finiteness of its set of residuals. They only differ for infinite-state systems, such as labeled Petri nets. [Recent diagnosability results could be found in \[1, 33, 7, 19\], etc.](#)

Based on condition (A'), we consider a new type of detectability, which we call *eventual strong detectability*. Formally, eventual strong detectability implies that for every infinite output sequence σ generated by a system, there exists a positive integer k_σ such that each prefix σ' of σ with length greater than k_σ allows reconstructing the current state. We will prove that eventual strong detectability is strictly weaker than strong detectability and strictly stronger than weak detectability, for labeled Petri nets and even for deterministic finite automata satisfying Assumption 1.

We will also prove that eventual strong detectability can be verified in polynomial time for finite automata. For labeled Petri nets, we show that the property is decidable and the corresponding decision problem is EXPSPACE-hard: note that this decidability result holds under the *promptness* assumption (collected in (ii) of Assumption 2) that is actually equivalent to condition (ii) of Assumption 1 for labeled Petri nets.

Approximate detectability

State estimation is usually a preliminary step that a plant operator must address so that, depending on the state value, a suitable action may be taken, [which is also similar to the state disambiguation problem in the literature \[42, 26, 35\]](#). Examples include computing a control input in supervisory control, raising an alarm in fault diagnosis, inferring a secret in an opacity problem, reacting to the detection of a cyber-attack, etc. The number of these possible actions is usually finite and this naturally determines a finite partition of the system's state space into equivalence classes, each one corresponding to states for which the same action should be taken. In such a context, it is not

necessary to solve a detectability problem, i.e., determine the exact value of the state, but just to solve an approximate version of it, i.e., determine to which class the state belongs.

The notion of approximate detectability applies to all previously defined detectability notions, weak or strong. Here we just study one of them, namely *weak approximate detectability* which implies that, given a finite partition of the state space, there exists an integer k and an infinite output sequence generated by a system each of whose prefixes of length greater than k allows determining the partition cell to which the current state belongs. In this paper, we will prove that weak approximate detectability is undecidable for labeled P/T nets. For finite automata, we will prove that deciding this property is PSPACE-complete. The undecidable result is obtained by reducing the undecidable language equivalence problem for labeled P/T nets to negation of the weak approximate detectability problem. The result for finite automata is obtained by using related results for weak detectability of finite automata [44, 30].

1.3 Paper structure

To help the reader better understand the contribution of the paper, the relations among the different detectability properties studied in this work are shown in Tabs. 1 and 2. The table also includes known results on strong detectability and weak detectability of finite automata and labeled Petri nets proved in [17, 44].

The remainder of the paper is as follows. Section 2 introduces necessary preliminaries, including finite automata, labeled Petri nets, the language equivalence problem, and the coverability problem, together with necessary tools such as Dickson's lemma, Yen's path formulae, etc. Section 3 collects the results on weak approximate detectability for finite automata and labeled Petri nets. Section 4 consists of the results on eventual strong detectability also for both models. Section 5 ends up with a short conclusion. We first study weak approximate detectability because fewer tools are needed than in studying eventual strong detectability.

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 set S , S^* and S^ω are used to denote the sets of finite sequences (called *words*) of elements of S including the empty word ϵ and infinite sequences (called *configurations*) of elements of S , respectively. As usual, we denote $S^+ = S^* \setminus \{\epsilon\}$. For a word $s \in S^*$, $|s|$ 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^ω denote the k -length word and configuration consisting of copies of s 's, respectively. For a word (configuration) $s \in S^*(S^\omega)$, a word $s' \in S^*$ is called a *prefix* of s , denoted as $s' \sqsubset s$, if there exists another word (configuration) $s'' \in S^*(S^\omega)$ 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. For a word $s \in S^*$, where $S = \{s_1, \dots, s_n\}$, $\#(s)(s_i)$ denotes the number of s_i 's occurrences in s , $i \in [1, n]$.

Strong detectability	\Rightarrow	Eventual strong detectability
decidable	\neq	decidable
([17])	(Fig. 4)	(Thm. 4)
EXPSPACE-hard		EXPSPACE-hard
([17])		(Thm. 4)
		\Downarrow
		\nexists (Fig. 9)
Weak approximate detectability	\Leftarrow	Weak detectability
undecidable	\neq	undecidable
(Thm. 1)	(Fig. 2)	([17])

Table 1: Relationships among different detectability notions for labeled Petri nets, where \Rightarrow means “the notion on the tail is stronger than the notion on the head”, \neq means “the two notions are not equivalent by the counterexample shown in the figure beside \neq ”, the decidability result for strong detectability proved in [17] is based on Assumption 1, and can be strengthened to hold only based on the promptness assumption which is actually (ii) of Assumption 1 for labeled Petri nets by using our proposed extended concurrent composition method similarly as in the proof of Theorem 4. The decidability result for eventual strong detectability is also based on the promptness assumption.

Strong detectability	\Rightarrow	Eventual strong detectability
<hr/> P	\neq	<hr/> $O(s^4e^2)$
([27])	(Fig. 4)	(Thm. 3)
		\Downarrow
		\nVdash (Fig. 9)
Weak approximate detectability	\Leftarrow	Weak detectability
<hr/> PSPACE-complete	\neq	<hr/> PSPACE-complete
(Thm. 2)	(Fig. 2)	([44])

Table 2: Relationships among different detectability notions for finite automata, where s and e are the numbers of states and events, \Rightarrow means “the notion on the tail is stronger than the notion on the head”, \neq means “the two notions are not equivalent by the counterexample shown in the figure beside \neq ”; the polynomial-time verification algorithm for strong detectability given in [27] applies to finite automata satisfying Assumption 1, but generally does not apply to finite automata that violate Assumption 1; by using the proposed method in our paper, one can design a polynomial-time verification algorithm for strong detectability of all finite automata; the exponential-time verification algorithm for weak detectability given in [27] actually applies to finite automata satisfying the assumption of non-emptiness of generated ω -languages that is weaker than Assumption 1, and in this paper we characterize how to verify the weaker assumption and how to deal with the case when the weaker assumption is not satisfied.

A partition of a set S is a set of nonempty subsets of S such that these subsets are pairwise disjoint and their union equals S .

2.1 Labeled state-transition systems

In order to formulate detectability notions in a uniform manner, we introduce *labeled state-transition systems* (LSTSs) as follows, which contain finite automata and labeled Petri nets as special cases. An LSTS is formulated as a sextuple

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

where X is a set of states, T a set of events, $X_0 \subset X$ a set of initial states, $\rightarrow \subset X \times T \times X$ a transition relation, Σ a set of outputs (labels), and $\ell : T \rightarrow \Sigma \cup \{\epsilon\}$ a *labeling* function. As usual, we use $\ell^{-1}(\sigma)$ to denote the *preimage* $\{t \in T \mid \ell(t) = \sigma\}$ of an output $\sigma \in \Sigma$. A state $x \in X$ is called *deadlock* if $(x, t, x') \notin \rightarrow$ 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*. 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 $t \in T$, 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 $\ell(t_1 t_2 \dots) = \ell(t_1) \ell(t_2) \dots$ and $\ell(\epsilon) = \epsilon$. For all $x, x' \in X$ and $t \in T$, we also denote $x \xrightarrow{t} x'$ if $(x, t, x') \in \rightarrow$. More generally, we denote all transitions $x \xrightarrow{t_1} x_1, x_1 \xrightarrow{t_2} x_2, \dots, x_{n-1} \xrightarrow{t_n} x_n$ by $x \xrightarrow{t_1 \dots t_n} x_n$ for short, where n is a positive integer. We say a state $x' \in X$ is *reachable from a state* $x \in X$ if there exist $t_1, \dots, t_n \in T$ such that $x \xrightarrow{t_1 \dots t_n} x'$, where n is a positive integer. We say a subset 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 subset* 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 some initial state. For an LSTS \mathcal{S} , we call the new LSTS the *accessible part* (denoted by $\text{Acc}(\mathcal{S})$) of \mathcal{S} that is obtained from \mathcal{S} by removing all non-reachable states. An LSTS \mathcal{S} is called *deterministic* if for all $x, x', x'' \in X$ and all $t \in T$, if $(x, t, x') \in \rightarrow$ and $(x, t, x'') \in \rightarrow$ then $x' = x''$.

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

2.2 Finite automata

A DES can be modeled by a finite automaton or a labeled Petri net. In order to represent a DES, we consider a *finite automaton* as a finite LSTS $\mathcal{S} = (X, T, X_0, \rightarrow, \Sigma, \ell)$, i.e., when X, T, Σ are finite. Such a finite automaton is also obtained from a standard finite automaton [31] by removing all accepting states, replacing a unique initial state by a set X_0 of initial states, and adding a labeling function ℓ . *In the sequel, a finite automaton always means a finite LSTS.* Transitions $x \xrightarrow{t} x'$ with $\ell(t) = \epsilon$ are called ϵ -*transitions* (or *unobservable transitions*), and other transitions are called *observable transitions*.

2.3 Labeled Petri nets

A *net* is a quadruple $N = (P, T, Pre, Post)$, where P is a finite set of *places* graphically represented by circles; T is a finite set of *transitions* graphically represented by bars; $P \cup T \neq \emptyset$, $P \cap T = \emptyset$; $Pre : P \times T \rightarrow \mathbb{N}$ and $Post : P \times T \rightarrow \mathbb{N}$ are the *pre-* and *post-incidence functions* that specify the arcs directed from places to transitions, and vice versa. Graphically $Pre(p, t)$ is the weight of the arc $p \rightarrow t$ and $Post(p, t)$ is the weight of the arc $t \rightarrow p$ for all $(p, t) \in P \times T$. The *incidence function* is defined as $C = Post - Pre$.

A *marking* is a map $M : P \rightarrow \mathbb{N}$ that assigns to each place of a net a natural number of tokens, graphically represented by black dots. For a marking $M \in \mathbb{N}^P$, the restriction of M to a subset P' of P is denoted by $M|_{P'}$. For a marking $M \in \mathbb{N}^P$, a transition $t \in T$ is called *enabled* at M if $M(p) \geq Pre(p, t)$ for all $p \in P$, and is denoted by $M[t]$, where as usual \mathbb{N}^P denotes the set of maps from P to \mathbb{N} . An enabled transition t at M may *fire* and yield a new making $M'(p) = M(p) + C(p, t)$ for all $p \in P$, written as $M[t]M'$. As usual, we assume that at each marking and each time step, at most one transition fires. For a marking M , a sequence $t_1 \dots t_n$ of transitions is called *enabled* at M if t_1 is enabled at M , t_2 is enabled at the unique M_2 satisfying $M[t_1]M_2$, \dots , t_n is enabled at the unique M_{n-1} satisfying $M[t_1] \dots [t_{n-1}]M_{n-1}$. We write the firing of $t_1 \dots t_n$ at M as $M[t_1 \dots t_n]$ for short, and similarly denote the firing of $t_1 \dots t_n$ at M yielding M' by $M[t_1 \dots t_n]M'$. $\mathcal{T}(N, M_0) := \{s \in T^* | M_0[s]\}$ is used to denote the set of transition sequences enabled at M_0 . Particularly we have $M_0[\epsilon]M_0$. A pair (N, M_0) is called a *Petri net* or a *place/transition net (P/T net)*, where $N = (P, T, Pre, Post)$ is a net, $M_0 : P \rightarrow \mathbb{N}$ is called the *initial marking*, and the Petri net evolves initially at M_0 as transition sequences fire. Denote the set of *reachable markings* of the Petri net by $\mathcal{R}(N, M_0) := \{M \in \mathbb{N}^P | \exists s \in T^*, M_0[s]M\}$.

A *labeled P/T net* is a quadruple (N, M_0, Σ, ℓ) , where N is a net, M_0 is an initial marking, Σ is an *alphabet* (a finite set of labels), and $\ell : T \rightarrow \Sigma \cup \{\epsilon\}$ is a *labeling function* that assigns to each transition $t \in T$ a symbol of Σ or the empty word ϵ , which means when a transition t fires, its label $\ell(t)$ can be observed if $\ell(t) \in \Sigma$; and nothing can be observed if $\ell(t) = \epsilon$. A transition $t \in T$ is called *observable* if $\ell(t) \in \Sigma$, and called *unobservable* otherwise. Particularly, a labeling function $\ell : T \rightarrow \Sigma$ is called ϵ -*free*, and a P/T net with an ϵ -free labeling function is called an ϵ -*free labeled P/T net*. A Petri net is actually an ϵ -free labeled P/T net with an injective labeling function. For a labeled P/T net $G = (N, M_0, \Sigma, \ell)$, the *language generated by G* is

denoted by $\mathcal{L}(G) := \{\sigma \in \Sigma^* \mid \exists s \in T^*, M_0[s], \ell(s) = \sigma\}$, i.e., the set of labels of finite transition sequences enabled at the initial marking M_0 . We also say for each $\sigma \in \mathcal{L}(G)$, G generates σ . For $\sigma \in \Sigma^\omega$, we say G generates σ if an infinite event sequence $t_1 t_2 \dots \in T^\omega$ is enabled at M_0 (denoted $M_0[t_1 t_2 \dots]$) and $\ell(t_1 t_2 \dots) = \sigma$. The set of infinite label sequences generated by G is denoted by $\mathcal{L}^\omega(G)$ (which is an ω -language).

Note that for a labeled P/T net $G = (N, M_0, \Sigma, \ell)$, when we observe a label sequence $\sigma \in \Sigma^*$, there may exist infinitely many firing transition sequences labeled by σ . However, for an ϵ -free labeled P/T net, when we observe a label sequence σ , there exist at most finitely many firing transition sequences labeled by σ . Denote by $\mathcal{M}(G, \sigma) := \{M \in \mathbb{N}^P \mid \exists s \in T^*, M_0[s]M, \ell(s) = \sigma\}$, the set of markings in which G can be when σ is observed. Then for each $\sigma \in \Sigma^*$, $\mathcal{M}(G, \sigma)$ is finite for an ϵ -free labeled P/T net G .

2.4 The language equivalence problem

The undecidable result proved in this paper is obtained by using the following language equivalence problem.

Proposition 1 [10, Theorem 8.2] *It is undecidable to verify whether two ϵ -free labeled P/T nets with the same alphabet generate the same language.*

2.5 Dickson's lemma

Let P be a finite set. For every two elements x and y of \mathbb{N}^P , we say $x \leq y$ if and only if $x(p) \leq y(p)$ for all p in P . We write $x < y$ if $x \leq y$ and $x \neq y$. For a subset S of \mathbb{N}^P , an element $x \in S$ is called *minimal* if for all y in S , $y \leq x$ implies $y = x$. Dickson's lemma [5] shows that for each subset S of \mathbb{N}^P , there exist at most finitely many distinct minimal elements. This lemma follows from the fact that every infinite sequence with all elements in \mathbb{N}^P has an increasing infinite subsequence, where such an increasing subsequence can be chosen component-wise [22, Theorem 2.5]. We will use Dickson's lemma to prove some decidable results for labeled P/T nets.

2.6 The coverability problem

We also need the following Proposition 2 on the coverability problem to obtain some main results on complexity.

Proposition 2 [21, 15] *It is EXPSpace-complete to decide for a Petri net $G = (N, M_0)$ and a destination marking $M \in \mathbb{N}^P$ whether G covers M , i.e., whether there exists a marking $M' \in \mathcal{R}(N, M_0)$ such that $M \leq M'$.*

In [15], it is proved that deciding coverability for Petri nets requires at least 2^{cn} space infinitely often for some constant $c > 0$, where n is the number of transitions. In [21], it is shown that deciding this property for a Petri net requires at most space $2^{cm \log m}$ for some constant c , where m is the size of the set of all transitions. For a Petri net $((P, T, Pre, Post), M_0)$, each transition $t \in T$ corresponds to a $|P|$ -length vector $Post(\cdot, t) - Pre(\cdot, t) =: c(t)$ whose components are integers. The size of t is the sum of the lengths of the binary representations of the components of $c(t)$ (where the length of 0 is 1). The size of T is the sum of the sizes of all transitions of T , and is set to be the above m .

The coverability problem belongs to EXPSPACE [21]. Proposition 2 has been used to prove the EXPSPACE-hardness of checking diagnosability [40] and prognosability [39] of labeled Petri nets.

2.7 Infinite graphs

Let (V, E) be a *directed graph*, where V is the *vertex set*, and $E \subset V \times V$ the *edge set*. For each edge $(v, v') \in E$, also denoted by $v \rightarrow v'$, v and v' are called the *tail* and the *head* of the edge, respectively, v is called a *parent* of v' and v' is called a *child* of v . A directed graph is called *infinite* if it has infinitely many vertices. A *path* is a sequence of vertices connected by edges with the same direction, i.e., a path is of one of the forms: (1) $\cdots \rightarrow v_{-1} \rightarrow v_0 \rightarrow v_1 \rightarrow \cdots$ (bi-infinite), (2) $v_0 \rightarrow v_1 \rightarrow \cdots$ (infinite), (3) $\cdots \rightarrow v_{-1} \rightarrow v_0$ (anti-infinite), or (4) $v_1 \rightarrow \cdots \rightarrow v_n$ (finite). For each finite path $v_1 \rightarrow \cdots \rightarrow v_n$, v_1 is called an *ancestor* of v_n , and v_n is called a *descendant* of v_1 . A directed graph (V, E) is called a *tree* if there is a vertex v_0 without any parent (called *root*), any other vertex is a descendant of v_0 and the head of exactly one edge. A tree is called *locally finite* if each vertex has at most finitely many children.

2.8 Yen's path formulae for Petri nets

The final tool that we will use to prove some decidable results is Yen's path formula [37, 2] for Petri nets. In [37], a concept of Yen's path formulae is proposed and some upper bounds for verifying the satisfiability of the formulae are studied. In addition, it is shown that many problems, e.g., the boundedness problem, the coverability problem for Petri nets, can be reduced to the satisfiability problem of some Yen's path formulae. In [2], a special class of Yen's path formulae called *increasing* Yen's path formulae is proposed. The main results of [2] are stated as follows.

Proposition 3 ([2]) *The reachability problem for Petri nets can be reduced to the satisfiability problem of some Yen's path formula, and the satisfiability problem of each Yen's path formula can be reduced to the reachability problem for Petri nets with respect to the marking with all places empty, all in polynomial time. In addition, the satisfiability of each increasing Yen's path formula can be verified in EXPSPACE.*

For a Petri net (N, M_0) , where $N = (P, T, Pre, Post)$ is a net, each Yen's path formula consists of the following elements:

1. *Variables.* There are two types of variables, namely, *marking variables* M_1, M_2, \dots and *variables for transition sequences* s_1, s_2, \dots , where each M_i denotes an indeterminate function in \mathbb{Z}^P and each s_i denotes an indeterminate finite sequence of transitions, \mathbb{Z} is the set of integers.
2. *Terms.* Terms are defined recursively as follows.
 - (a) \forall constant $c \in \mathbb{N}^P$, c is a term.
 - (b) $\forall j > i$, $M_j - M_i$ is a term, where M_i and M_j are marking variables.
 - (c) $T_1 + T_2$ and $T_1 - T_2$ are terms if T_1 and T_2 are terms.
3. *Atomic Predicates.* There are two types of atomic predicates, namely *transition predicates* and *marking predicates*.
 - (a) *Transition predicates.*
 - $y \odot \#(s_i) < c$, $y \odot \#(s_i) = c$, and $y \odot \#(s_i) > c$ are predicates, where $i > 1$, constant $y \in \mathbb{Z}^T$, constant $c \in \mathbb{N}$, and \odot denotes the inner product (i.e., $(a_1, \dots, a_{|T|}) \odot (b_1, \dots, b_{|T|}) = \sum_{i=1}^{|T|} a_i b_i$).
 - $\#(s_1)(t) \leq c$ and $\#(s_1)(t) \geq c$ are predicates, where constant $c \in \mathbb{N}$, $t \in T$.
 - (b) *Marking predicates.*
 - **Type 1.** $M(p) \geq c$ and $M(p) > c$ are predicates, where M is a marking variable and $c \in \mathbb{Z}$ is constant.
 - **Type 2.** $T_1(i) = T_2(j)$, $T_1(i) < T_2(j)$, and $T_1(i) > T_2(j)$ are predicates, where T_1, T_2 are terms and $i, j \in T$.
4. $F_1 \vee F_2$ and $F_1 \wedge F_2$ are predicates if F_1 and F_2 are predicates.

A Yen's path formula f is of the following form (with respect to Petri net (N, M_0) , where $N = (P, T, Pre, Post)$):

$$(\exists M_1, \dots, M_n \in \mathbb{N}^P)(\exists s_1, \dots, s_n \in T^*)[(M_0[s_1]M_1[s_2] \cdots [s_n]M_n) \wedge F(M_1, \dots, M_n, s_1, \dots, s_n)], \quad (1)$$

where $F(M_1, \dots, M_n, s_1, \dots, s_n)$ is a predicate.

Given a Petri net G and a Yen's path formula f , we use $G \models f$ to denote that f is true in G . The satisfiability problem is the problem of determining, given a Petri net G and a Yen's path formula f , whether $G \models f$.

A Yen's path formula (1) is called *increasing* if F does not contain transition predicates and implies $M_n \geq M_1$. When $n = 1$, it naturally holds $M_n \geq M_1$, then in this case an increasing Yen's path formula is $(\exists M_1)(\exists s_1)[(M_0[s_1]M_1) \wedge F(M_1)]$.

The unboundedness problem can be formulated as the satisfiability of the increasing Yen's path formula $(\exists M_1, M_2)(\exists s_1, s_2)[(M_0[s_1]M_1[s_2]M_2) \wedge (M_2 > M_1)]$.

The coverability problem can be formulated as the satisfiability of the increasing Yen's path formula $(\exists M_1)(\exists s_1)[(M_0[s_1]M_1) \wedge (M_1 \geq M)]$, where M is the destination marking.

3 Weak approximate detectability

The concept of weak detectability is formulated as follows.

Definition 1 (WD) Consider an LSTS $\mathcal{S} = (X, T, X_0, \rightarrow, \Sigma, \ell)$. System \mathcal{S} is called weakly detectable if $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$ implies there exists a label sequence $\sigma \in \mathcal{L}^\omega(\mathcal{S})$ such that for some positive integer k , $|\mathcal{M}(\mathcal{S}, \sigma')| = 1$ for every prefix σ' of σ satisfying $|\sigma'| \geq k$.

Sometimes, we do not need to determine the current state of an LSTS, but only need to know whether the current state belongs to some prescribed subset of reachable states. Then the concept of weak approximate detectability is formulated as below.

Definition 2 (WAD) Consider an LSTS $\mathcal{S} = (X, T, X_0, \rightarrow, \Sigma, \ell)$. Given a positive integer $n > 1$ and a partition $\{R_1, \dots, R_n\}$ of the set of its reachable states, \mathcal{S} is called weakly approximately detectable with respect to partition $\{R_1, \dots, R_n\}$ if $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$ implies there exists a label sequence $\sigma \in \mathcal{L}^\omega(\mathcal{S})$ such that for some positive integer k , for every prefix σ' of σ satisfying $|\sigma'| \geq k$, $\emptyset \neq \mathcal{M}(\mathcal{S}, \sigma') \subset R_{i_{\sigma'}}$, for some $i_{\sigma'} \in [1, n]$.

3.1 Labeled Petri nets

One directly sees that if an LSTS is weakly detectable, then it is weakly approximately detectable with respect to every finite partition of its state space. However, if it is weakly approximately detectable with respect to some finite partition of its state space, then it is not necessarily weakly detectable. See the following example.

Example 1 Consider a labeled Petri net G in Fig. 2. We have $\mathcal{L}^\omega(G) = \{a^\omega, b^\omega\}$. We also have for all $k \in \mathbb{Z}_+$, $\mathcal{M}(G, a^k) = \{(0, 1, 0, 0, 0), (1, 0, 0, 0, 0)\}$, $\mathcal{M}(G, b^k) = \{(0, 0, 0, 1, 0), (0, 0, 0, 0, 1)\}$, where the components of a marking is in the order $(p_{-2}, p_{-1}, p_0, p_1, p_2)$. These observations show that the net is not weakly detectable. It is weakly approximately detectable with respect to the partition:

$$\begin{aligned} R_1 &= \{(0, 0, 1, 0, 0)\}, \\ R_2 &= \{(0, 0, 0, 1, 0), (0, 0, 0, 0, 1)\}, \\ R_3 &= \{(0, 1, 0, 0, 0), (1, 0, 0, 0, 0)\} \end{aligned} \tag{2}$$

of the set of its reachable markings. Also, this net is a nondeterministic finite automaton with $(0, 0, 1, 0, 0)$ being the unique initial state. Similarly we have the automaton is also weakly approximately detectable with respect to partition (2) but not weakly detectable. In addition, this net becomes a deterministic finite automaton if we regard a and b as labels of four different events, respectively, and the corresponding deterministic finite automaton is also weakly approximately detectable with respect to partition (2) but not weakly detectable.

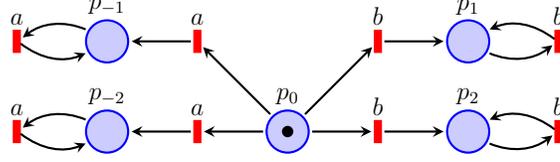


Figure 2: A labeled P/T net G , where letters beside transitions denote their labels, each arc is with weight 1.

For the weak approximate detectability of labeled P/T nets, the following result holds.

Theorem 1 *Let $n > 1$ be a positive integer. It is undecidable to verify for an ϵ -free labeled P/T net and a partition $\{R_1, \dots, R_n\}$ of the set of its reachable markings, whether the labeled P/T net is weakly approximately detectable with respect to $\{R_1, \dots, R_n\}$.*

Proof We prove this result by reducing the language equivalence problem of labeled Petri nets (Proposition 1) to the problem under consideration. We only need to prove the case $n = 2$, since the undecidability of the case for any n greater than 2 trivially follows from that. In addition, in our reduction, the partition is computable by using the reachability algorithm ([13, 18, 14]).

Arbitrarily given two ϵ -free labeled P/T nets $G_i = (N_i, M_0^i, \Sigma, \ell_i)$, where $N_i = (P_i, T_i, Pre_i, Post_i)$, $i = 1, 2$, $P_1 \cap P_2 = \emptyset$, $T_1 \cap T_2 = \emptyset$, we next construct a new ϵ -free labeled P/T net $G = (N_G, M_0^G, \Sigma \cup \{\sigma_G\}, \ell_G)$ from G_1 and G_2 . G is specified as follows: (1) Add 5 places $p_0, p_1^1, p_1^2, p_2, p_3$ to G_1 and G_2 , where initially p_0 has one token, and all the other places have no token. (2) Add 6 transitions $t_0^1, t_0^2, t_1^1, t_1^2, t_2, t_3$, and arcs $p_0 \rightarrow t_0^1 \rightarrow p_1^1 \rightarrow t_1^1 \rightarrow p_2 \rightarrow t_2 \rightarrow p_3 \rightarrow t_3 \rightarrow p_2$, and $p_0 \rightarrow t_0^2 \rightarrow p_1^2 \rightarrow t_1^2 \rightarrow p_3$, where these transitions are labeled by $\sigma_G \notin \Sigma$. (3) For each transition $t \in T_i$, add arcs $p_1^i \rightarrow t \rightarrow p_1^i$, $i = 1, 2$. (4) All these newly added arcs are with weight 1. See Fig. 3 as a sketch.

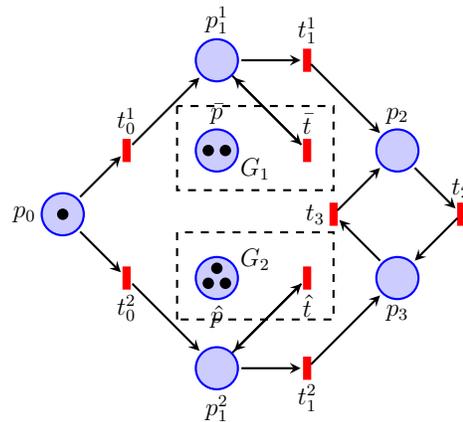


Figure 3: Sketch for the reduction in the proof of Theorem 1, where all transitions outside $G_1 \cup G_2$ are with the same label.

For net G , initially only transition t_0^1 or t_0^2 can fire. After t_0^1 (t_0^2) fires, the unique token in place p_0 moves to place p_1^1 (p_1^2), initializing net G_1 (G_2). While G_1 (G_2) is running, only transition t_1^1 (t_1^2) outside $T_1 \cup T_2$ can fire. The firing of t_1^1 (t_1^2) moves the token in place p_1^1 (p_1^2) to place p_2 (p_3), and terminates the running of G_1 (G_2), yielding that the token in p_2 (p_3) can move along the direction $p_2 \rightarrow p_3 \rightarrow p_2$ periodically forever, but G_1 (G_2) will never run again. Hence net G may fire only infinite transition sequences $t_0^1 s t_1^1 (t_2 t_3)^\omega$, $t_0^1 s'$, $t_0^2 r t_1^2 (t_3 t_2)^\omega$, or $t_0^2 r'$, where $s \in (T_1)^*$, $s' \in (T_1)^\omega$, $r \in (T_2)^*$, $r' \in (T_2)^\omega$. So G can generate only configurations $\sigma_G \sigma (\sigma_G)^\omega$ or $\sigma_G \sigma'$ where $\sigma \in \Sigma^*$, $\sigma' \in \Sigma^\omega$. Note that for some nets G_1 and G_2 , the corresponding net G never fires $t_0^1 s'$ or $t_0^2 r'$ as above, e.g., when $\mathcal{L}(G_1) \cup \mathcal{L}(G_2)$ is finite; but for all G_1 and G_2 , the corresponding G fires $t_0^1 s t_1^1 (t_2 t_3)^\omega$ and $t_0^2 r t_1^2 (t_3 t_2)^\omega$ as above.

We partition the set $\mathcal{R}(N_G, M_0^G)$ of reachable markings of net G as follows:

$$\begin{aligned} R_1 &= \{M \in \mathbb{N}^{P_G} \mid M(p_0) \text{ or } M(p_1^1) \text{ or } M(p_2) = 1, M(p_1^2) = M(p_3) = 0\} \\ &\quad \cap \mathcal{R}(N_G, M_0^G), \\ R_2 &= \{M \in \mathbb{N}^{P_G} \mid M(p_1^2) \text{ or } M(p_3) = 1, M(p_0) = M(p_1^1) = M(p_2) = 0\} \\ &\quad \cap \mathcal{R}(N_G, M_0^G). \end{aligned} \tag{3}$$

By using the reachability algorithm in the literature, one can decide whether an arbitrary given marking belongs to R_1 , R_2 , or neither R_1 nor R_2 .

If $\mathcal{L}(G_1) \neq \mathcal{L}(G_2)$, without loss of generality, we assume that there exists $\sigma \in \mathcal{L}(G_1) \setminus \mathcal{L}(G_2)$. Then when G generates configuration $\sigma_G \sigma (\sigma_G)^\omega$, it can fire only transition sequences $t_0^1 s t_1^1 (t_2 t_3)^\omega$, where $s \in (T_1)^*$, $\ell_G(s) = \sigma$. It can be directly seen for each positive integer k , $\emptyset \neq \mathcal{M}(G, \sigma_G \sigma (\sigma_G)^k) \subset R_{k \bmod 2 + 1}$, where $k \bmod 2$ means the remainder of k divided by 2. That is, net G is weakly approximately detectable with respect to partition (3).

Next we assume that $\mathcal{L}(G_1) = \mathcal{L}(G_2)$. Note that net G generates only configurations $\sigma_G \sigma'$ or $\sigma_G \sigma (\sigma_G)^\omega$, where $\sigma' \in \Sigma^\omega$, $\sigma \in \Sigma^*$. For the former case, for each prefix σ'' of σ' , there exist firing sequences $s \in (T_1)^*$ of net G_1 and $r \in (T_2)^*$ of net G_2 such that $\ell_G(s) = \ell_G(r) = \sigma''$, and markings $M_G, M'_G \in \mathbb{N}^{P_G}$ such that $M_0^G [t_0^1 s] M_G$, $M_0^G [t_0^2 r] M'_G$, $M_G(p_1^1) = 1$, $M_G(p_1^2) = 0$, $M'_G(p_1^1) = 0$, and $M'_G(p_1^2) = 1$, then we have $\mathcal{M}(G, \sigma'') \cap R_1 \neq \emptyset$ and $\mathcal{M}(G, \sigma'') \cap R_2 \neq \emptyset$. For the latter case, chosen an arbitrary prefix $\sigma_G \sigma (\sigma_G)^k$ of $\sigma_G \sigma (\sigma_G)^\omega$, where k is an arbitrary positive integer, we have there exist firing sequences $s \in (T_1)^*$ of net G_1 and $r \in (T_2)^*$ of net G_2 such that $\ell_G(s) = \ell_G(r) = \sigma$ and net G can fire both $t_0^1 s s'$ and $t_0^2 r r'$, where s' and r' are k length prefixes of $(t_2 t_3)^\omega$ and $(t_3 t_2)^\omega$, respectively. Since G will fire both $t_0^1 s s'$ and $t_0^2 r r'$, we have $\mathcal{M}(G, \sigma_G \sigma (\sigma_G)^k) \cap R_1 \neq \emptyset$ and $\mathcal{M}(G, \sigma_G \sigma (\sigma_G)^k) \cap R_2 \neq \emptyset$. Hence for each positive integer k , $\mathcal{M}(G, \sigma_G \sigma (\sigma_G)^k)$ intersects both R_1 and R_2 . We have checked all label sequences generated by G , hence G is not weakly approximately detectable with respect to partition (3).

□

3.2 Finite automata

Next, we study the complexity of deciding weak approximate detectability of finite automata.

An exponential-time algorithm for verifying weak detectability of a finite automaton \mathcal{S} under Assumption 1 is given in [27], but the algorithm actually applies to every \mathcal{S} satisfying $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$ which is weaker than Assumption 1. Automaton \mathcal{S} satisfying $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$ is naturally weakly detectable and hence weakly approximately detectable with respect to very fine partition of its set of reachable states as well, and the condition $\mathcal{L}^\omega(\mathcal{S}) = \emptyset$ can be verified in linear time of the size of \mathcal{S} by computing all strongly connected components of \mathcal{S} . Note that in Assumption 1, (ii) is actually a little weaker than the counterpart in [30, 27], as in these two papers, there is no requirement “reachable from an initial state”. However, one easily sees that existence of a cycle not reachable from an initial state consisting of only unobservable events does not violate the verification results for weak detectability given in [27].

Assumption 1 *An LSTS $\mathcal{S} = (X, T, X_0, \rightarrow, \Sigma, \ell)$ satisfies*

- (i) *\mathcal{S} is deadlock-free,*
- (ii) *no cycle in \mathcal{S} reachable from an initial state contains only unobservable events, i.e., for every reachable state $x \in X$ and every nonempty unobservable event sequence s , there exists no transition sequence $x \xrightarrow{s} x$ in \mathcal{S} .*

In Assumption 1, (i) guarantees that the automaton never halts, (ii) ensures that for each infinite event sequence generated by the automaton, the corresponding label sequence is also of infinite length.

It is not difficult to see that weak approximate detectability is PSPACE-complete for finite automata. In order to show the PSPACE-hardness of weak approximate detectability with respect to a partition of cardinality n , we can slightly change the reduction in our paper [44] to reduce the finite automaton intersection problem to weak approximate detectability in polynomial time. To prove the PSPACE membership, we can reduce weak approximate detectability to weak detectability in polynomial time by constructing a quotient automaton from the original automaton, where elements of the corresponding partition are states of the quotient automaton. Hence the following theorem holds.

Theorem 2 1. *The weak approximate detectability of finite automata can be verified in PSPACE.*

- 2. *Deciding weak approximate detectability of deterministic finite automata whose events can be directly observed is PSPACE-hard.*

Remark 1 *The notion of weak approximate detectability can be extended from a finite partition of the set of reachable states to a finite cover of that set. Such an extension may have potential applications in supervisor reduction of supervisory control theory. In supervisory control theory,*

the optimal solution to the control problem associated with a DES is the supremal supervisor (the supremal controllable sublanguage), and it is important to reduce the size of the supremal supervisor together with preserving some corresponding control actions [4, 32, 34], where the reduction is done based on a notion of control cover that is actually a cover of the state set. Under this extension, it is not difficult to see that the extended weak approximate detectability of finite automata can also be verified in PSPACE by the powerset construction used to verify weak detectability in [27], and it is undecidable to verify this notion for labeled Petri nets (from Theorem 1).

4 Eventual strong detectability

The concepts of strong detectability and eventual strong detectability are given as follows. The former implies there exists a positive integer k such that for each infinite label sequence generated by a system, each prefix of the label sequence of length greater than k allows reconstructing the current state. The latter implies that for each infinite label sequence generated by a system, there exists a positive integer k (depending on the label sequence) such that each prefix of the label sequence of length greater than k allows doing that. Hence the former is stronger than the latter.

Definition 3 (SD) Consider an LSTS $\mathcal{S} = (X, T, X_0, \rightarrow, \Sigma, \ell)$. System \mathcal{S} is called strongly detectable if there exists a positive integer k such that for each label sequence $\sigma \in \mathcal{L}^\omega(\mathcal{S})$, $|\mathcal{M}(\mathcal{S}, \sigma')| = 1$ for every prefix σ' of σ satisfying $|\sigma'| > k$.

Definition 4 (ESD) Consider an LSTS $\mathcal{S} = (X, T, X_0, \rightarrow, \Sigma, \ell)$. System \mathcal{S} is called eventually strongly detectable if for each label sequence $\sigma \in \mathcal{L}^\omega(\mathcal{S})$, there exists a positive integer k_σ such that $|\mathcal{M}(\mathcal{S}, \sigma')| = 1$ for every prefix σ' of σ satisfying $|\sigma'| > k_\sigma$.

By definition, strong detectability implies eventual strong detectability. The following Proposition 4 shows that they are not equivalent.

Proposition 4 Strong detectability strictly implies eventual strong detectability for labeled P/T nets and finite automata.

Proof Consider the labeled P/T net G in Fig. 4, where a and b are labels of transitions. It can be seen that $\mathcal{L}^\omega(G) = a^\omega + a^*b^\omega + a^*ba^\omega := \{a^\omega\} \cup \{a^n b^\omega | n \in \mathbb{N}\} \cup \{a^n b a^\omega | n \in \mathbb{N}\}$. One also has that $\mathcal{M}(G, a^n) = \{(1, 0, 0)\}$, $\mathcal{M}(G, a^n b) = \{(0, 1, 0), (0, 0, 1)\}$, $\mathcal{M}(G, a^n b b^{m+1}) = \{(0, 1, 0)\}$, $\mathcal{M}(G, a^n b a^{m+1}) = \{(0, 0, 1)\}$ for all $m, n \in \mathbb{N}$. Hence G is eventually strongly detectable, but not strongly detectable.

The net can be regarded as a deterministic finite automaton satisfying Assumption 1 when a and b are regarded as labels of events. By a direct observation, it is also eventually strongly detectable, but not strongly detectable.

□

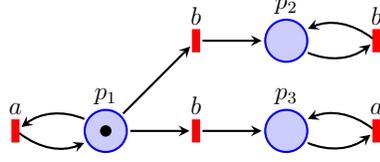


Figure 4: A labeled P/T net G that is eventually strongly detectable, but not strongly detectable.

4.1 Finite automata

In order to give an easily understandable way to verify eventual strong detectability of finite automata, for a finite automaton $\mathcal{S} = (X, T, X_0, \rightarrow, \Sigma, \ell)$, we next construct three new automata from \mathcal{S} .

Firstly, we construct its concurrent composition

$$\text{CC}_A(\mathcal{S}) = (X', T', X'_0, \rightarrow') \quad (4)$$

as follows:

1. $X' = X \times X$;
2. $T' = T'_o \cup T'_\epsilon$, where $T'_o = \{(\check{t}, \check{t}') \mid \check{t}, \check{t}' \in T, \ell(\check{t}) = \ell(\check{t}') \in \Sigma\}$, $T'_\epsilon = \{(\check{t}, \epsilon) \mid \check{t} \in T, \ell(\check{t}) = \epsilon\} \cup \{(\epsilon, \check{t}) \mid \check{t} \in T, \ell(\check{t}) = \epsilon\}$;
3. $X'_0 = X_0 \times X_0$;
4. for all $(\check{x}_1, \check{x}'_1), (\check{x}_2, \check{x}'_2) \in X'$, $(\check{t}, \check{t}') \in T'_o$, $(\check{t}'', \epsilon) \in T'_\epsilon$, and (ϵ, \check{t}''') $\in T'_\epsilon$,
 - $((\check{x}_1, \check{x}'_1), (\check{t}, \check{t}'), (\check{x}_2, \check{x}'_2)) \in \rightarrow'$ if and only if $(\check{x}_1, \check{t}, \check{x}_2), (\check{x}'_1, \check{t}', \check{x}'_2) \in \rightarrow$,
 - $((\check{x}_1, \check{x}'_1), (\check{t}'', \epsilon), (\check{x}_2, \check{x}'_2)) \in \rightarrow'$ if and only if $(\check{x}_1, \check{t}'', \check{x}_2) \in \rightarrow, \check{x}'_1 = \check{x}'_2$,
 - $((\check{x}_1, \check{x}'_1), (\epsilon, \check{t}'''), (\check{x}_2, \check{x}'_2)) \in \rightarrow'$ if and only if $\check{x}_1 = \check{x}_2, (\check{x}'_1, \check{t}''', \check{x}'_2) \in \rightarrow$.

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, $\text{CC}_A(\mathcal{S})$ aggregates every pair of transition sequences of \mathcal{S} producing the same label sequence. In addition, $\text{CC}_A(\mathcal{S})$ has at most $|X|^2$ states and at most $|X|^2(2|T_\epsilon||X| + \sum_{\sigma \in \Sigma} |\ell^{-1}(\sigma)|^2|X|^2)$ transitions, where the number does not exceed $|X|^2(2|T_\epsilon||X| + |T_o|^2|X|^2)$. Hence it takes time $O(2|X|^3|T_\epsilon| + |X|^4 \sum_{\sigma \in \Sigma} |\ell^{-1}(\sigma)|^2)$ to construct $\text{CC}_A(\mathcal{S})$. For the special case when all observable events can be directly observed studied in [27], the complexity reduces to $O(2|X|^3|T_\epsilon| + |X|^4|T_o|)$. See the following example.

Example 2 A finite automaton \mathcal{S} and its concurrent composition $\text{CC}_A(\mathcal{S})$ are shown in Fig. 5.

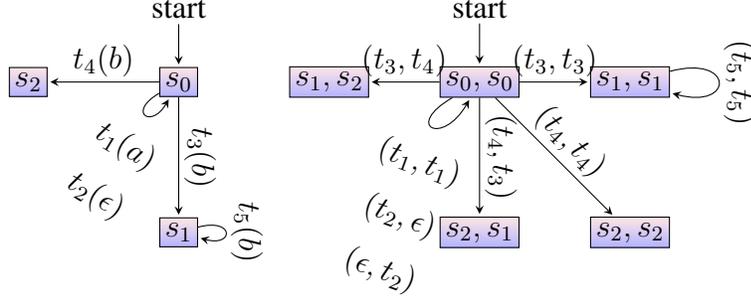


Figure 5: A finite automaton (left) and its concurrent composition (right, only the accessible part illustrated).

Secondly, we construct its *observation automaton*

$$\text{Obs}(\mathcal{S}) = (X, \{\varepsilon, \hat{\varepsilon}\}, X_0, \rightarrow', \{\hat{\varepsilon}\}, \ell') \quad (5)$$

in linear time of the size of \mathcal{S} , where $\rightarrow' \subset X \times \{\varepsilon, \hat{\varepsilon}\} \times X$, $\ell'(\varepsilon) = \varepsilon$, $\ell'(\hat{\varepsilon}) = \hat{\varepsilon}$, for every two states $x, x' \in X$, $(x, \hat{\varepsilon}, x') \in \rightarrow'$ 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 for all $t' \in T$ with $(x, t', x') \in \rightarrow$, $\ell(t') = \varepsilon$. Here the labeling function ℓ is also naturally extended to $\ell' : \{\varepsilon, \hat{\varepsilon}\}^* \cup \{\varepsilon, \hat{\varepsilon}\}^\omega \rightarrow \{\hat{\varepsilon}\}^* \cup \{\hat{\varepsilon}\}^\omega$. One sees that $\mathcal{L}^\omega(\mathcal{S}) \neq \emptyset$ if and only if in $\text{Obs}(\mathcal{S})$ there is a transition sequence $x_0 \xrightarrow{s} x \xrightarrow{s'} x$ such that $x_0 \in X_0$, $s, s' \in \{\varepsilon, \hat{\varepsilon}\}^*$, and $\ell'(s') \neq \varepsilon$.

Thirdly, we also need to construct its *bifurcation automaton*

$$\text{Bifur}(\mathcal{S}) = (X, \{\bar{\varepsilon}, \check{\varepsilon}\}, X_0, \rightarrow', \{\bar{\varepsilon}, \check{\varepsilon}\}, \ell') \quad (6)$$

in linear time of the size of \mathcal{S} , where $\rightarrow' \subset X \times \{\bar{\varepsilon}, \check{\varepsilon}\} \times X$, $\ell'(\bar{\varepsilon}) = \bar{\varepsilon}$, $\ell'(\check{\varepsilon}) = \check{\varepsilon}$, ℓ' is also naturally extended to $\ell' : \{\bar{\varepsilon}, \check{\varepsilon}\}^* \cup \{\bar{\varepsilon}, \check{\varepsilon}\}^\omega \rightarrow \{\bar{\varepsilon}, \check{\varepsilon}\}^* \cup \{\bar{\varepsilon}, \check{\varepsilon}\}^\omega$, transitions $x \xrightarrow{\bar{\varepsilon}} x'$ are called *fair transitions*, transitions $x \xrightarrow{\check{\varepsilon}} x'$ are called *bifurcation transitions*, for every two states $i, j \in X$, (1) $(j, \bar{\varepsilon}, i), (j, \check{\varepsilon}, i) \notin \rightarrow'$ if $\neg A_1$, (2) $(x, \bar{\varepsilon}, x') \in \rightarrow'$ if $A_1 \wedge A_2 \wedge A_3$, (3) $(x, \check{\varepsilon}, x') \in \rightarrow'$ otherwise, where

$$\begin{aligned} A_1 &= (\exists t \in T)[(j, t, i) \in \rightarrow], \\ A_2 &= (\nexists t \in T, j' \in X)[((j, t, j') \in \rightarrow) \wedge (\ell(t) = \varepsilon) \wedge (j' \neq j)], \\ A_3 &= (\forall t \in T)[(((j, t, i) \in \rightarrow) \wedge (\ell(t) \neq \varepsilon)) \implies \\ &\quad (\nexists t' \in T, j' \in X)[((j, t', j') \in \rightarrow) \wedge (\ell(t') = \ell(t)) \wedge (j' \neq i)]]. \end{aligned}$$

One sees that both fair transitions and bifurcation transitions can be ε -transitions or observable transitions. Next we explain the relation between $\text{Bifur}(\mathcal{S})$, the original automaton \mathcal{S} , and the concurrent composition $\text{CC}_A(\mathcal{S})$. Here (1) holds if there is no transition from state j to state i in \mathcal{S} ; (2) holds if there exists a transition from j to i , and none of such transitions has a bifurcation in \mathcal{S} ; and (3) holds if there is a transition from j to i that has a bifurcation also in \mathcal{S} . For the case that (3) holds, if A_1 holds but A_2 does not hold, then for \mathcal{S} one has $\{j\} \subsetneq \mathcal{M}(\{j\}, \varepsilon)$ and hence $|\mathcal{M}(\{j\}, \varepsilon)| > 1$, for $\text{CC}_A(\mathcal{S})$ there is a transition $(j, j) \xrightarrow{(\varepsilon, \bar{t})} (j, i')$ with $\ell(\bar{t}) = \varepsilon$ and $i' \neq j$; if A_1

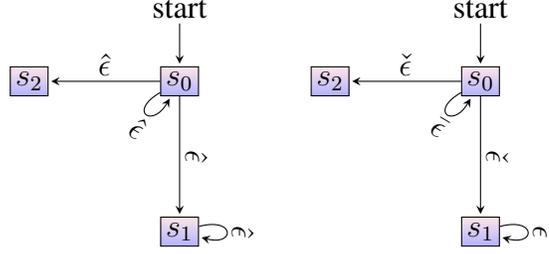


Figure 6: Observation automaton (left) and bifurcation automaton (right) of the automaton in the left part of Fig. 5.

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 $(j, \tilde{t}', i) \in \rightarrow$; for $\text{CC}_A(\mathcal{S})$ there is a transition $(j, j) \xrightarrow{(\tilde{t}', \tilde{t}'')} (i, i')$ with $i' \neq i$ and $\ell(\tilde{t}') = \ell(\tilde{t}'')$ for the above \tilde{t}' .

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 $\text{Obs}(\mathcal{S})$ if and only if there is a transition from x to x' in $\text{Bifur}(\mathcal{S})$. This obvious observation is helpful in verifying eventual strong detectability for finite automata.

Example 3 *Reconsider the finite automaton \mathcal{S} in Example 2 (in the left part of Fig. 5). Its observation automaton and bifurcation automaton are seen in Fig. 6. It has a unique initial state and generates a nonempty ω -language. In addition, all its states are reachable.*

We next use the concurrent composition, the observation automaton, and the bifurcation automaton of a finite automaton \mathcal{S} defined by (4), (5), and (6) to verify its eventual strong detectability without any assumption. Note that by using a similar way, one can design a polynomial-time algorithm for verifying strong detectability, which strengthens the polynomial-time verification algorithm given in [27] under Assumption 1. In addition, the method in [27] can also be used to check eventual strong detectability, but also only under Assumption 1.

Theorem 3 *The eventual strong detectability of finite automata can be verified in polynomial time.*

Proof Consider a finite automaton $\mathcal{S} = (X, T, X_0, \rightarrow, \Sigma, \ell)$ and another finite automaton $\text{Acc}(\text{CC}_A(\mathcal{S})) = (X', T', X'_0, \rightarrow')$. We use $\text{Acc}(\text{CC}_A(\mathcal{S}))$, $\text{Obs}(\text{Acc}(\mathcal{S}))$, and $\text{Bifur}(\text{Acc}(\mathcal{S}))$ to verify its eventual strong detectability.

One observes by definition that \mathcal{S} is not eventually strongly detectable if and only if

$$\text{there is an infinite transition sequence } x_0 \xrightarrow{s_1} \text{ such that} \quad (7a)$$

$$x_0 \in X_0, \ell(s_1) \in \Sigma^\omega \text{ and for every } n \in \mathbb{Z}_+, \text{ there is a prefix} \quad (7b)$$

$$s'_1 \text{ of } s_1 \text{ satisfying } |\ell(s'_1)| > n \text{ and } |\mathcal{M}(\mathcal{S}, \ell(s'_1))| > 1. \quad (7c)$$

We claim that (7) holds if and only if one of the following items holds:

(1) In $\text{Acc}(\text{CC}_A(\mathcal{S}))$, there exists an infinite transition sequence

$$x'_0 \xrightarrow{s'_1} x'_1 \xrightarrow{s'_2} \dots \quad (8)$$

such that $x'_i \in X'_0$, for every $i \in \mathbb{Z}_+$, $s'_i \in (T')^*$, $\ell(s'_i) \in \Sigma^+$, and $s'_i(L) \neq s'_i(R)$.

(2) In \mathcal{S} , there exists an infinite transition sequence

$$x_0 \xrightarrow{s_1} x_1 \xrightarrow{s_2} x_2 \xrightarrow{s_3} \dots \quad (9)$$

such that $x_0 \in X_0$, for all $i \in \mathbb{Z}_+$, $s_i \in T^*$, $\ell(s_{i+1}) \in \Sigma^+$, and $|\mathcal{M}(\{x_i\}, \sigma)| > 1$ for some $\sigma \sqsubset \ell(s_{i+1})$.

It is trivial to see that either Item (1) or Item (2) implies (7).

Conversely suppose that (7) holds but Item (2) does not hold. Then for \mathcal{S} , there is an infinite transition sequence

$$\bar{x}_0 \xrightarrow{\bar{s}_1} \bar{x}_1 \xrightarrow{\bar{s}_2} \bar{x}_2 \xrightarrow{\bar{s}_3} \dots \quad (10)$$

satisfying (7b) and (7c) such that for every $i \in \mathbb{Z}_+$, $\bar{s}_i \in T^*$, $\ell(\bar{s}_{i+1}) \in \Sigma^+$, and $|\mathcal{M}(\{\bar{x}_i\}, \bar{\sigma})| = 1$ for all $\bar{\sigma} \sqsubset \ell(\bar{s}_{i+1})$. Fix such a sequence (10). Then for every $i \in \mathbb{Z}_+$, there exists a finite transition sequence

$$\bar{x}_0^i \xrightarrow{\bar{s}_1^i} \dots \xrightarrow{\bar{s}_i^i} \bar{x}_i^i \quad (11)$$

such that $\bar{x}_0^i \in X_0$, for all $j \in [1, i]$, one has $\ell(\bar{s}_j^i) = \ell(\bar{s}_j)$, $\bar{x}_j^i \neq \bar{x}_j$. Choose k sufficiently large, by the finiteness of X , we obtain a transition sequence

$$\bar{x}'_0 \xrightarrow{\bar{s}'_1} \dots \xrightarrow{\bar{s}'_k} \bar{x}'_k \quad (12)$$

of $\text{Acc}(\text{CC}_A(\mathcal{S}))$ such that $\bar{x}'_0 \in X'_0$, the left component and the right component of (12) are a prefix of (10) and (11) with $i = k$; for all $i \in [1, k]$, $\bar{x}'_i(L) \neq \bar{x}'_i(R)$, and $\bar{x}'_{l'} = \bar{x}'_{l''}$ for some $0 < l' < l'' \leq k$. Then the prefix $\bar{x}'_0 \xrightarrow{\bar{s}'_1} \dots \xrightarrow{\bar{s}'_{l'}} \bar{x}'_{l'} \xrightarrow{\bar{s}'_{l'+1}} \dots \xrightarrow{\bar{s}'_{l''}} \bar{x}'_{l''}$ of (12) can be extended to an infinite transition sequence of the form (8) by repeating $\bar{x}'_{l'} \xrightarrow{\bar{s}'_{l'+1}} \dots \xrightarrow{\bar{s}'_{l''}} \bar{x}'_{l''}$ for infinitely many times, i.e., Item (1) holds.

Next we show that both Item (1) and Item (2) can be verified in polynomial time.

Observe that Item (1) holds if and only if in $\text{Acc}(\text{CC}_A(\mathcal{S}))$, there is a finite transition sequence

$$\tilde{x}'_0 \xrightarrow{\tilde{s}'_1} \tilde{x}'_1 \xrightarrow{\tilde{s}'_2} \tilde{x}'_1 \quad (13)$$

with $\tilde{x}'_0 \in X'_0$, $\tilde{s}'_1, \tilde{s}'_2 \in (T')^*$ such that $\ell(\tilde{s}'_2) \in \Sigma^+$ and $\tilde{x}'_1(L) \neq \tilde{x}'_1(R)$. Next we verify (13) in polynomial time. See Fig. 7 for a sketch.

1. Compute $\text{Obs}(\text{Acc}(\text{CC}_A(\mathcal{S})))$.

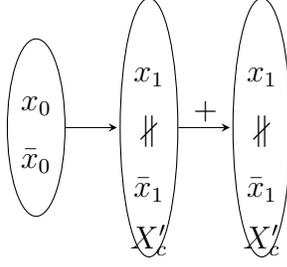


Figure 7: A sketch for verifying (13).

2. Compute all strongly connected components of $\text{Obs}(\text{Acc}(\text{CC}_A(\mathcal{S})))$.
3. Denote the set of states (x, \bar{x}) of $\text{Obs}(\text{Acc}(\text{CC}_A(\mathcal{S})))$ with $x \neq \bar{x}$ that belong to a cycle with nonempty label sequence by X'_c , check whether $X'_c \neq \emptyset$.

Each of the first two steps costs linear time of $\text{CC}_A(\mathcal{S})$. Note that $X'_c \neq \emptyset$ if and only if (13) holds. Observe that $X'_c \neq \emptyset$ if and only if in one of the obtained strongly connected components, there is an observable transition and a state (x', \bar{x}') with $x' \neq \bar{x}'$. Hence the third step also costs linear time. Overall, verifying Item (1) costs linear time of $\text{CC}_A(\mathcal{S})$, at most $O(|X|^4|T|^2)$.

Also observe that Item (2) holds if and only if in \mathcal{S} , there exists a finite transition sequence

$$\tilde{x}_0 \xrightarrow{\tilde{s}_1} \tilde{x}_1 \xrightarrow{\tilde{s}_2} \tilde{x}_1 \quad (14)$$

such that $\tilde{x}_0 \in X_0$, $\tilde{s}_1, \tilde{s}_2 \in T^*$, $\ell(\tilde{s}_2) \in \Sigma^+$, and $|\mathcal{M}(\{\tilde{x}_1\}, \sigma)| > 1$ for some $\sigma \sqsubset \ell(\tilde{s}_2)$.

Next we show that (14) can be verified in polynomial time. See Fig. 8 for a sketch.

1. Compute $\text{Obs}(\text{Acc}(\mathcal{S}))$ and $\text{Bifur}(\text{Acc}(\mathcal{S}))$.
2. Compute X_{oc} and X_{bc} , where X_{oc} (resp. X_{bc}) is the set of states of $\text{Acc}(\mathcal{S})$ that belong to a cycle containing an observable transition (resp. a bifurcation transition).
3. Check whether $X_{oc} \cap X_{bc} = \emptyset$.

Note that a state x of $\text{Acc}(\mathcal{S})$ belongs to a cycle containing an observable transition (resp. a bifurcation transition) if and only if x is any state of any strongly connected component of $\text{Obs}(\text{Acc}(\mathcal{S}))$ (resp. $\text{Bifur}(\text{Acc}(\mathcal{S}))$) that contains an observable transition (resp. a bifurcation transition). Then one has $X_{oc} \cap X_{bc} \neq \emptyset$ if and only if (14) holds. Hence it takes linear time of \mathcal{S} to check whether Item (2) holds.

□

Example 4 Recall the finite automaton \mathcal{S} in the left part of Fig. 5. Following the procedure in the proof of Theorem 3, by Figs. 5 and 6, we have $X_{oc} = \{s_0, s_1\}$, $X_{bc} = \emptyset$, $X_{oc} \cap X_{bc} = \emptyset$ (implying that Item (2) does not hold), and $X'_c = \emptyset$ (implying that Item (1) does not hold either), then \mathcal{S} is eventually strongly detectable.



$$X_{oc} \cap X_{bc} \quad X_{oc} \cap X_{bc}$$

Figure 8: A sketch for verifying (14).

Remark 2 By using a similar method as in the proof of Theorem 3, one can design a polynomial-time verification algorithm for strong detectability of finite automata without any assumption, with complexity linear of the size of $\text{Obs}(\text{Acc}(\text{CC}_A(\mathcal{S})))$, hence at most $O(|X|^4|T|^2)$.

Let us analyse the computational complexity of using [27, Theorem 5] to verify strong detectability of finite automata satisfying Assumption 1. In [27], for a finite automaton \mathcal{S} (satisfying Assumption 1), a nondeterministic finite automaton G_{det} with at most $|X|^2/2 + |X|/2 + 1$ states and at most $(|X|^2/2 + |X|/2 + 1)^2|T|$ transitions is constructed to verify its strong detectability, where every state of G_{det} is a subset of states of \mathcal{S} with cardinality 1 or 2, except for the initial state of G_{det} being a superset of X_0 . The time consumption for computing G_{det} is as follows:

$$|X| \quad \underbrace{|T_\epsilon||X|}_{\text{compute initial state } Q_0 \text{ of } G_{det} \text{ by traversing all } \epsilon\text{-transition sequences from } X_0, \text{ where } X_0 \subset Q_0 \subset X} \quad + \quad (15)$$

$$|Q_0| \underbrace{|T_o||X|}_{\text{compute } Q' \subset X \text{ by traversing all observable transitions from } Q_0} + \underbrace{|\Sigma||X||T_\epsilon||X|}_{\text{compute } Q'' \text{ by traversing all } \epsilon\text{-transition sequences from } Q'} + \underbrace{|\Sigma||X|^2}_{\text{split } Q'' \text{ into subsets of } X \text{ of cardinality 2 to obtain non-initial states of } G_{det}} + \quad (16)$$

$$\underbrace{|X|^2(2|T_o||X| + |\Sigma||X||T_\epsilon||X| + |\Sigma||X|^2)}_{\text{Repeat (16) from states of } G_{det} \text{ of cardinality 2}} + \quad (17)$$

$$\underbrace{|X|(|T_o||X| + |\Sigma||X||T_\epsilon||X| + |\Sigma||X|^2)}_{\text{Repeat (16) from states of } G_{det} \text{ of cardinality 1}}, \quad (18)$$

i.e., at most $O(2|X|^3|T_o| + |X|^4|\Sigma||T_\epsilon| + |X|^4|\Sigma|)$. For the special case when all observable events can be directly observed studied in [27], the complexity is $O(2|X|^3|T_o| + |X|^4|T_o||T_\epsilon| + |X|^4|T_o|)$. Actually, this construction tracks sets of states of \mathcal{S} with consistent observations, which is similar to the powerset construction that is of exponential size of \mathcal{S} . It is proved that \mathcal{S} is strongly detectable if and only if every state of G_{det} reachable from a cycle is a singleton. This condition can be checked in linear time of G_{det} by computing strongly connected components of G_{det} .

However, this method generally does not apply to a finite automaton that does not satisfy Assumption 1. For example, let us consider the finite automaton \mathcal{S} in the left part of Fig. 5. Remove the self-loop on s_1 , and denote the new automaton by $\bar{\mathcal{S}}$. Then one directly sees that $\mathcal{L}^\omega(\bar{\mathcal{S}}) = \{a^\omega\}$, and $\bar{\mathcal{S}}$ is strongly detectable. However, in the corresponding G_{det} , which consists of a self-loop with label a on $\{s_0\}$ and a transition from $\{s_0\}$ to $\{s_1, s_2\}$ with label b , there is a state $\{s_1, s_2\}$ with cardinality 2 reachable from a cycle, hence $\bar{\mathcal{S}}$ is not strongly detectable by [27, Theorem 5].

Actually, the verification method does not apply to this example because, two deadlock states s_1 and s_2 are not in any infinite-length transition sequence, but reachable from a state s_0 that belongs to an infinite-length transition sequence with infinite-length label sequence.

Remark 3 *Eventual strong detectability is a uniform concept. That is, a labeled Petri net is eventually strongly detectable if and only if it is eventually strongly detectable when its initial marking is replaced by any of its reachable markings. Formally, for a labeled Petri net $G = (N, M_0, \Sigma, \ell)$, G is eventually strongly detectable if and only if $G' = (N, M, \Sigma, \ell)$ is eventually strongly detectable for each $M \in \mathcal{R}(N, M_0)$.*

Example 5 *Consider a labeled P/T net G as shown in Fig. 9, where a, b are labels. We have $\mathcal{L}^\omega(G) = a^\omega + a^*b^\omega$, $|\mathcal{M}(G, a^n)| = 1$, $|\mathcal{M}(G, a^n b^m)| = 2$ for all $m, n \in \mathbb{Z}_+$. Hence the net is weakly detectable, but not eventually strongly detectable. The deterministic finite automaton obtained from the net when a and b are regarded as labels of events (particularly b as the label of four different events) is also weakly detectable, but not eventually strongly detectable.*

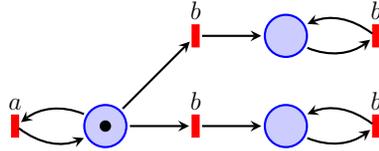


Figure 9: A labeled P/T net G that is weakly detectable, but not eventually strongly detectable.

4.2 Labeled Petri nets

In this subsection we discuss the decidability and complexity of eventual strong detectability for labeled Petri nets.

If a labeled Petri net G satisfies $\mathcal{L}^\omega(G) = \emptyset$, then it is naturally eventually strongly detectable. Actually whether the property $\mathcal{L}^\omega(G) = \emptyset$ holds can be verified in EXPSPACE, and can also be guaranteed by the following Assumption 2 that is weaker than the widely used Assumption 1 in detectability studies of DESs.

Proposition 5 *Verifying whether a labeled Petri net G satisfies $\mathcal{L}^\omega(G) = \emptyset$ belongs to EXPSPACE.*

Proof Consider a labeled Petri net $G = (N = (P, T, Pre, Post), M_0, \Sigma, \ell)$. Observe that $\mathcal{L}^\omega(G) \neq \emptyset$ if and only if there exists an infinite firing sequence

$$M_0[s_1]M_1[s_2] \cdots \quad (19)$$

such that for each $i \in \mathbb{Z}_+$, $\ell(s_i) \in \Sigma^+$.

For G , a sequence (19) exists if and only if G satisfies the following Yen's path formula

$$(\exists \widetilde{M}_1, \widetilde{M}_2)(\exists \widetilde{s}_1, \widetilde{s}_2)[(M_0[\widetilde{s}_1]\widetilde{M}_1[\widetilde{s}_2]\widetilde{M}_2) \wedge (\widetilde{M}_2 \geq \widetilde{M}_1) \wedge (\ell(\widetilde{s}_2) \in \Sigma^+)]. \quad (20)$$

The “if” part follows from $\widetilde{M}_1[\widetilde{s}_2]\widetilde{M}_2$ being a repetitive firing sequence (hence can consecutively fire for infinitely many times) and $|\ell(\widetilde{s}_2)| > 0$.

For the “only if” part: Arbitrarily fix a sequence (19). By Dickson's lemma, in the set $\{M_0, M_1, \dots\}$, there are totally finitely many distinct minimal elements. Choose $k > 0$ such that $\{M_0, \dots, M_k\}$ contains the maximal number of distinct minimal elements of $\{M_0, M_1, \dots\}$, then there exist $0 \leq k' \leq k < k''$ such that $M_{k'} \leq M_{k''}$. Then the firing sequence $M_0[s_1 \dots s_{k'}]M_{k''}[s_{k'+1} \dots s_{k''}]M'$ satisfies $M_{k'} \leq M'$ and $\ell(s_{k'+1} \dots s_{k''}) \in \Sigma^+$.

The satisfiability of (20) is actually a fair nondetermination problem and hence belongs to EXPSPACE [2, Subsection 6.1]. \square

Assumption 2 (i) *A labeled P/T net G does not terminate, i.e., there exists an infinite firing sequence at the initial marking, and*

(ii) *it is prompt, i.e., there exists no repetitive firing sequence labeled by the empty string.*

Note that the deadlock-freeness assumption (see (i) of Assumption 1) implies (i) of Assumption 2, but not vice versa; (ii) of Assumption 2 is actually equivalent to (ii) of Assumption 1 for labeled Petri nets. Note also that for a labeled P/T net G , $\mathcal{L}^\omega(G) \neq \emptyset$ implies that G does not terminate, but not vice versa, because transitions could be labeled by ϵ . Verifying termination of Petri nets (the first part of Assumption 2) is EXPSPACE-complete by the results of [21, 15]. Verifying promptness of labeled Petri nets belongs to EXPSPACE [2]. In addition, promptness is equivalent to all infinite firing sequences being labeled by infinite-length sequences.

In order to characterize eventual strong detectability for labeled Petri nets, we introduce the concurrent composition of a labeled Petri net. Given a labeled P/T net $G = (N = (P, T, Pre, Post), M_0, \Sigma, \ell)$, we construct in polynomial time its concurrent composition as a Petri net

$$CC_N(G) = (N' = (P', T', Pre', Post'), M'_0) \quad (21)$$

which aggregates every pair of firing sequences of G producing the same label sequence. Denote $P = \{\check{p}_1, \dots, \check{p}_{|P|}\}$ and $T = \{\check{t}_1, \dots, \check{t}_{|T|}\}$, duplicate them to $P_i = \{\check{p}_1^i, \dots, \check{p}_{|P|}^i\}$ and $T_i = \{\check{t}_1^i, \dots, \check{t}_{|T|}^i\}$, $i = 1, 2$, where we let $\ell(\check{t}_i^1) = \ell(\check{t}_i^2) = \ell(\check{t}_i)$ for all i in $[1, |T|]$. Then we specify G' as follows:

1. $P' = P_1 \cup P_2$;
2. $T' = T'_o \cup T'_e$, where $T'_o = \{(\check{t}_i^1, \check{t}_j^2) \in T_1 \times T_2 \mid i, j \in [1, |T|], \ell(\check{t}_i^1) = \ell(\check{t}_j^2) \in \Sigma\}$, $T'_e = \{(\check{t}_1, \epsilon) \mid \check{t}_1 \in T_1, \ell(\check{t}_1) = \epsilon\} \cup \{(\epsilon, \check{t}_2) \mid \check{t}_2 \in T_2, \ell(\check{t}_2) = \epsilon\}$;

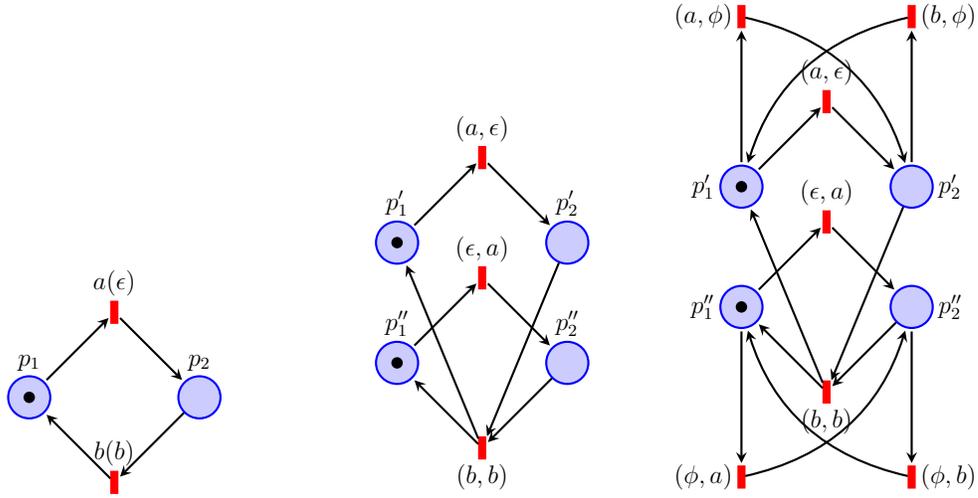


Figure 10: A labeled Petri net G , where event a is un-observable, but b can be directly observed. Figure 11: Concurrent composition of the net in Fig. 10. Figure 12: Extended concurrent composition of the net in Fig. 10.

3. for all $k \in [1, 2]$, all $l \in [1, |P|]$, and all $i, j \in [1, |T|]$ such that $\ell(\check{t}_i^1) = \ell(\check{t}_j^2) \in \Sigma$,

$$Pre'(\check{p}_l^k, (\check{t}_i^1, \check{t}_j^2)) = \begin{cases} Pre(\check{p}_l^k, \check{t}_i^1) & \text{if } k = 1, \\ Pre(\check{p}_l^k, \check{t}_j^2) & \text{if } k = 2, \end{cases}$$

$$Post'(\check{p}_l^k, (\check{t}_i^1, \check{t}_j^2)) = \begin{cases} Post(\check{p}_l^k, \check{t}_i^1) & \text{if } k = 1, \\ Post(\check{p}_l^k, \check{t}_j^2) & \text{if } k = 2; \end{cases}$$

4. for all $l \in [1, |P|]$, all $i \in [1, |T|]$ such that $\ell(\check{t}_i^1) = \ell(\check{t}_i^2) = \epsilon$,

$$Pre'(\check{p}_l^1, (\check{t}_i^1, \epsilon)) = Pre(\check{p}_l^1, \check{t}_i^1),$$

$$Pre'(\check{p}_l^2, (\epsilon, \check{t}_i^2)) = Pre(\check{p}_l^2, \check{t}_i^2),$$

$$Post'(\check{p}_l^1, (\check{t}_i^1, \epsilon)) = Post(\check{p}_l^1, \check{t}_i^1),$$

$$Post'(\check{p}_l^2, (\epsilon, \check{t}_i^2)) = Post(\check{p}_l^2, \check{t}_i^2);$$

5. $M'_0(\check{p}_l^k) = M_0(\check{p}_l)$ for any k in $[1, 2]$ and any l in $[1, |P|]$.

A labeled Petri net and its concurrent composition are shown in Fig. 10 and Fig. 11, respectively.

Assume that there exists a label sequence $\sigma \in \mathcal{L}(G)$ such that $|\mathcal{M}(G, \sigma)| > 1$, then there exist transitions $t_{\mu_1}, \dots, t_{\mu_n}, t_{\omega_1}, \dots, t_{\omega_n} \in T \cup \{\epsilon\}$, where $n \geq 1$, such that $\ell(t_{\mu_i}) = \ell(t_{\omega_i})$ for all

$i \in [1, n]$, $\ell(t_{\mu_1} \dots t_{\mu_n}) = \ell(t_{\omega_1} \dots t_{\omega_n}) = \sigma$, $M_0[t_{\mu_1} \dots t_{\mu_n}]M_1$ and $M_0[t_{\omega_1} \dots t_{\omega_n}]M_2$ for different M_1 and M_2 both in \mathbb{N}^P . Then for $\text{CC}_N(G)$, we have $M'_0[(t_{\mu_1}^1, t_{\omega_1}^2) \dots (t_{\mu_n}^1, t_{\omega_n}^2)]M'$, where $M'(\check{p}_l^k) = M_k(\check{p}_l)$, $k \in [1, 2]$, $l \in [1, |P|]$, and $M'(\check{p}_{l'}^1) \neq M'(\check{p}_{l'}^2)$ for some $l' \in [1, |P|]$ (briefly denoted by $M'|_{P_1} \neq M'|_{P_2}$).

Assume that for each label sequence $\sigma \in \mathcal{L}(G)$, we have $|\mathcal{M}(G, \sigma)| = 1$. Then for all $M' \in \mathcal{R}(N', M'_0)$, $M'(\check{p}_l^1) = M'(\check{p}_l^2)$ for each l in $[1, |P|]$ (briefly denoted by $M'|_{P_1} = M'|_{P_2}$).

We next characterize eventual strong detectability for labeled P/T nets. If a labeled Petri net G satisfies $\mathcal{L}^\omega(G) = \emptyset$, then it is eventually strongly detectable.

Checking strong detectability for labeled P/T nets is proved to be decidable and EXPSPACE-hard in the size of a labeled P/T net [17] under Assumption 1 (it is not difficult to see that the assumption “there does not exist an infinite unobservable sequence” used in [17] is equivalent to promptness by Dickson’s lemma). Here the size of a P/T net $G = (N = (P, T, Pre, Post), M_0)$ is $\lceil \log |P| \rceil + \lceil \log |T| \rceil +$ the size of $\{Pre(p, t) | p \in P, t \in T\} \cup \{Post(p, t) | p \in P, t \in T\} \cup \{M_0(p) | p \in P\}$, where the last term means the sum of the lengths of the binary representations of the elements of $\{Pre(p, t) | p \in P, t \in T\} \cup \{Post(p, t) | p \in P, t \in T\} \cup \{M_0(p) | p \in P\}$ [2, 37]. Hence the size of a labeled P/T net can be defined as the sum of the size of its underlying P/T net and that of its labeling function $\ell : T \rightarrow \Sigma \cup \{\epsilon\}$, where the latter is actually no greater than $|T|$.

Consider a labeled Petri net G . Consider a reachable marking M_1 of G and a firing sequence $\psi = M_1[t_2]M_2[t_3] \dots [t_l]M_l$, where $l > 1$, t_i is a transition of G for every $i \in [2, l]$. We say that ψ has a bifurcation if there exists $k \in [2, l]$ such that in the concurrent composition $\text{CC}_N(G)$ of G , there is a firing sequence $M'_1[t'_2]M'_2[t'_3] \dots [t'_n]M'_n$ for some $n > 1$ and with all t'_2, \dots, t'_n being transitions of $\text{CC}_N(G)$ such that $M'_1|_{P_1} = M'_1|_{P_2} = M_1$, $M'_n|_{P_1} = M_k$, the left component of $t'_2 \dots t'_n$ equals $t_2 \dots t_k$, and $M'_{k'}|_{P_1} \neq M'_{k'}|_{P_2}$ for some $k' \in [2, n]$.

For G , for two infinite firing sequences

$$M_0[\tilde{t}_1]\widetilde{M}_1[\tilde{t}_2] \dots \text{ and} \quad (22a)$$

$$M_0[\widehat{t}_1]\widehat{M}_1[\widehat{t}_2] \dots, \quad (22b)$$

where $\tilde{t}_i, \widehat{t}_i$ are transitions of G for all $i \in \mathbb{Z}_+$, we call *they merge after a finite time* if in $\text{CC}_N(G)$, there is an infinite firing sequence $M'_0[t'_1]M'_1[t'_2] \dots$ with t'_1, t'_2, \dots all being transitions of $\text{CC}_N(G)$ such that the left component and right component of $t'_1 t'_2 \dots$ equal $\tilde{t}_1 \tilde{t}_2 \dots$ and $\widehat{t}_1 \widehat{t}_2 \dots$, respectively, and there exists $k \in \mathbb{Z}_+$ such that $M'_j|_{P_1} = M'_j|_{P_2}$ for all $j > k$.

Theorem 4 (1) *The eventual strong detectability of a labeled P/T net G under (ii) of Assumption 2 is decidable.*

(2) *Deciding whether a labeled P/T net G with $\mathcal{L}^\omega(G) \neq \emptyset$ is eventually strongly detectable is EXPSPACE-hard.*

Proof (1) Proof of the decidability result:

By Proposition 5, we first verify whether G satisfies $\mathcal{L}^\omega(G) \neq \emptyset$ in EXPSPACE. If no, then G is eventually strongly detectable. Otherwise, continue the following procedure.

Consider a labeled Petri net $G = (N = (P, T, Pre, Post), M_0, \Sigma, \ell)$ with $\mathcal{L}^\omega(G)$ being nonempty. By definition, G is not eventually strongly detectable if and only if there exists $\sigma \in \mathcal{L}^\omega(G)$ such that for all $k \in \mathbb{N}$ there exists a prefix $\bar{\sigma}$ of σ satisfying $|\bar{\sigma}| > k$ and $|\mathcal{M}(G, \bar{\sigma})| > 1$. We construct the concurrent composition $CC_N(G) = (N' = (P', T', Pre', Post'), M'_0)$ of G as in (21).

We claim that G is not eventually strongly detectable if and only if one of the following two items holds (see Examples 6 and 7):

(1) In $CC_N(G)$, there exists an infinite firing sequence

$$M'_0[s'_1]M'_1[s'_2] \cdots, \quad (23)$$

where for every $i \in \mathbb{Z}_+$, s'_i contains a transition of T'_o , and $M'_i|_{P_1} \neq M'_i|_{P_2}$.

(2) In G , there exists an infinite firing sequence

$$M_0[s_1]M_1[s_2]M_2[s_3] \cdots \quad (24)$$

such that $M_0[s_1]M_1$ has a bifurcation, for each $i \in \mathbb{Z}_+$, $\ell(s_i) \in \Sigma^+$, and $M_i[s_{i+1}]M_{i+1}$ also has a bifurcation.

Apparently if Item (1) or Item (2) holds, then G is not eventually strongly detectable.

Suppose that G is not eventually strongly detectable. Then there exists an infinite firing sequence

$$M_0[\bar{s}_1]\bar{M}_1[\bar{s}_2]\bar{M}_2[\bar{s}_3] \cdots \quad (25)$$

such that $\ell(\bar{s}_i) \in \Sigma^+$ and $|\mathcal{M}(G, \ell(\bar{s}_1 \dots \bar{s}_i))| > 1$ for all $i \in \mathbb{Z}_+$. Next we fix such a sequence (25).

Furthermore, suppose that Item (1) does not hold. Then (25) and each infinite firing sequence of G starting at M_0 and having the same label sequence as (25) will merge after a finite time, since the label sequence of (25) is of infinite length. Next we prove that Item (2) holds. If in (25), infinitely many of $M_0[\bar{s}_1]\bar{M}_1$, $\bar{M}_1[\bar{s}_2]\bar{M}_2$, \dots have bifurcations, then (25) is a firing sequence satisfying the requirement in Item (2). Next we assume that there are only finitely many of them having bifurcations, and reach a contradiction. Without loss of generality, we assume that only $M_0[\bar{s}_1]\bar{M}_1$ has a bifurcation. Then for each $k \in \mathbb{Z}_+$, there exists a firing sequence $M_0[\tilde{s}_k]\tilde{M}_k$ such that $\ell(\tilde{s}_k) \sqsubset \ell(\bar{s}_1\bar{s}_2 \dots)$, $|\ell(\tilde{s}_k)| > k$, and some prefix of (25) and $M_0[\tilde{s}_k]\tilde{M}_k$ can be combined to obtain a firing sequence $M'_0[\tilde{s}'_k]\tilde{M}'_k$ of $CC_N(G)$ such that the label sequence of the right component of \tilde{s}'_k equals $\ell(\tilde{s}_k)$, $\tilde{M}'_k|_{P_2} = \tilde{M}_k$, and $\tilde{M}'_k|_{P_1} \neq \tilde{M}'_k|_{P_2}$. Collecting all such firing sequences $M_0[\tilde{s}_k]\tilde{M}_k$, $k \in \mathbb{Z}_+$, we obtain a locally finite, infinite tree \mathbb{T} with M_0 the root. Also collect all such markings \tilde{M}_k , $k \in \mathbb{Z}_+$, to obtain a set \mathbb{M} . Observe that in \mathbb{T} , M_0 has infinitely many descendants of \mathbb{M} . Also observe in \mathbb{T} that one of the finitely many children of M_0 also has infinitely many descendants of \mathbb{M} , denote such a child of M_0 by \widehat{M}_1 , then we obtain a firing sequence $M_0[\widehat{t}_1]\widehat{M}_1$ of G , where $\widehat{t}_1 \in T$. Since \mathbb{T} is locally finite, repeating the process of looking for $M_0[\widehat{t}_1]\widehat{M}_1$, we can obtain an infinite firing sequence

$$M_0[\widehat{t}_1]\widehat{M}_1[\widehat{t}_2] \cdots \quad (26)$$

of G such that for each $i \in \mathbb{Z}_+$, \widehat{M}_i has infinitely many descendants of \mathbb{M} in \mathbb{T} . By (ii) of Assumption 2, we have (26) is labeled by an infinite-length label sequence. Also, since for each $i \in \mathbb{Z}_+$, $M_0[\widehat{t}_1 \dots \widehat{t}_i] \widehat{M}_i$ is a prefix of some path of \mathbb{T} , we have $\ell(\widehat{t}_1 \widehat{t}_2 \dots) = \ell(\bar{s}_1 \bar{s}_2 \dots)$. Then it is not difficult to see that (26) and (25) can be combined into an infinite firing sequence of $\text{CC}_N(G)$ satisfying the requirement in Item (1), which is a contradiction.

Next we prove that the satisfiability of Item (1) or Item (2) are both decidable, completing the proof of the decidability result of eventual strong detectability.

For Item (1):

We claim that Item (1) holds if and only if there exists a firing sequence

$$M'_0[s'_1]M'_1[s'_2]M'_2 \quad (27)$$

in $\text{CC}_N(G)$ satisfying

$$(M'_2 \geq M'_1) \wedge (s'_2 \text{ contains a transition in } T'_o) \wedge (M'_2|_{P_1} \neq M'_2|_{P_2}), \quad (28)$$

where $T'_o \subset T'$ is shown in (21). That is, we next prove that Item (1) holds if and only if

$$(\exists M'_1, M'_2)(\exists s'_1, s'_2)[(27) \wedge (28)] \quad (29)$$

is satisfied.

“if”: Assume that for $\text{CC}_N(G)$, Eqn. (29) holds. Then Item (1) holds, because $M'_2|_{P_1} \neq M'_2|_{P_2}$, s'_2 contains a transition in T'_o (hence $\ell(s'_2)$ is of positive length), and $M'_1[s'_2]M'_2$ is a repetitive firing sequence and can fire consecutively for infinitely many times.

“only if”: Assume that Item (1) holds, and fix a sequence (23).

By Dickson’s lemma, the set $\{M'_0, M'_1, \dots\}$ contains at most finitely many distinct minimal elements. Then there exists $k \in \mathbb{Z}_+$ such that $\{M'_0, \dots, M'_k\}$ contains the maximal number of distinct minimal elements of $\{M'_0, M'_1, \dots\}$. Hence there exists $0 \leq l \leq k$ such that $M'_l \leq M'_{k+1}$. Then the firing sequence

$$M'_0[s'_1 \dots s'_l]M'_l[s'_{l+1} \dots s'_{k+1}]M'_{k+1}$$

satisfies that $M'_{k+1} \geq M'_l$, $s'_{l+1} \dots s'_{k+1}$ contains at least one transition of T'_o , and $M'_{k+1}|_{P_1} \neq M'_{k+1}|_{P_2}$, i.e., (29) holds.

In (28), “ $M'_2 \geq M'_1$ ” can be expressed as combination of marking predicates, “ s'_2 contains a transition in T'_o ” is a transition predicate, only “ $M'_2|_{P_1} \neq M'_2|_{P_2}$ ” is not a predicate.

Add two new places p''_0 and p''_1 into $\text{CC}_N(G)$, where initially p''_0 contains exactly 1 token, but p''_1 contains no token; add one new transition r''_1 , and arcs $p''_0 \rightarrow r''_1 \rightarrow p''_1$, both with weight 1. Also, for each transition t in $\text{CC}_N(G)$, add arcs $p''_1 \rightarrow t \rightarrow p''_1$, both with weight 1. Then we obtain a new Petri net $\text{CC}_N(G)'$. We have $\text{CC}_N(G)$ satisfies (29) if and only if $\text{CC}_N(G)'$ satisfies the Yen’s path

formula

$$\begin{aligned}
& (\exists M_1'', M_2'', M_3'')(\exists s_1'', s_2'', s_3'')[\\
& (M_0''[s_1'']M_1''[s_2'']M_2''[s_3'']M_3'') \wedge \\
& (s_1'' = r_1'') \wedge (M_3'' \geq M_2'') \wedge (s_3'' \text{ contains a transition of } T_o'') \wedge \\
& ((M_3'' - M_1'')|_{P_1} \neq (M_3'' - M_1'')|_{P_2})],
\end{aligned} \tag{30}$$

where note that one always has $M_1''|_{P_1} = M_1''|_{P_2}$.

Then by Proposition 3, the satisfiability of (29) is decidable, implying that the satisfiability of Item (1) is decidable.

Next we prove that the satisfiability of Item (2) is decidable.

We claim that for G , Item (2) holds if and only if

$$\text{there exists a firing sequence } M_0[s_1]M_1[s_2]M_2 \text{ satisfying} \tag{31a}$$

$$\underline{M}_1 \leq \underline{M}_2, \tag{31b}$$

$$\underline{s}_2 \text{ contains an observable transition, and} \tag{31c}$$

$$\underline{M}_1[s_2]M_2 \text{ contains a bifurcation.} \tag{31d}$$

Assume that for G , Item (2) holds. Again by Dickson's lemma, there exist $0 \leq l < k$ such that the firing sequence $M_0[s_1 \dots s_l]M_l[s_{l+1} \dots s_k]M_k$ satisfies that $M_l \leq M_k$, $s_{l+1} \dots s_k$ contains an observable transition, and $M_l[s_{l+1} \dots s_k]M_k$ has a bifurcation. That is, (31) holds.

Assume that (31) holds. By (31b), (31c), and (31d), the sequence in (31a) can be extended to an infinite firing sequence

$$M_0[s_1]M_1[s_2]M_2[s_2](M_2 + (M_2 - M_1))[s_2] \cdots [s_2](M_2 + k(M_2 - M_1))[s_2] \cdots$$

satisfying for each $l \in \mathbb{Z}_+$, one has $(M_2 + l(M_2 - M_1))[s_2](M_2 + (l+1)(M_2 - M_1))$ has a bifurcation. That is, Item (2) holds.

Add a new set

$$T_\phi = T_\phi^1 \cup T_\phi^2$$

of transitions into $CC_N(G)$, where $\phi \notin T_1 \cup T_2$, $T_\phi^1 = \{(\check{t}_1, \phi) | \check{t}_1 \in T_1\}$, $T_\phi^2 = \{(\phi, \check{t}_2) | \check{t}_2 \in T_2\}$. Add the following rules: for all $l \in [1, |P|]$, all $i \in [1, |T|]$,

$$\begin{aligned}
Pre'(\check{p}_i^1, (\check{t}_i^1, \phi)) &= Pre(\check{p}_i^1, \check{t}_i^1), \\
Pre'(\check{p}_i^2, (\phi, \check{t}_i^2)) &= Pre(\check{p}_i^2, \check{t}_i^2), \\
Post'(\check{p}_i^1, (\check{t}_i^1, \phi)) &= Post(\check{p}_i^1, \check{t}_i^1), \\
Post'(\check{p}_i^2, (\phi, \check{t}_i^2)) &= Post(\check{p}_i^2, \check{t}_i^2).
\end{aligned}$$

The newly obtained extended concurrent composition is denoted by

$$CC_N^E(G) = (N'' = (P'', T'', Pre'', Post''), M_0''), \tag{32}$$

where $P'' = P'$, $T'' = T' \cup T_\phi$, $M_0'' = M_0'$. For example, the corresponding extended concurrent composition of the net in Fig. 10 is shown in Fig. 12.

Then for G , (31) holds if and only if for $\text{CC}_N^E(G)$,

there exists a firing sequence $M_0'''[s_1''']M_1'''[s_2''']M_2'''[s_3''']M_3'''$ such that (33a)

$$M_1''' \leq M_3''' , \quad (33b)$$

$$s_3''' \text{ contains a transition } (t_1, *) \text{ with } \ell(t_1) \in \Sigma, \quad (33c)$$

$$M_2'''|_{P_1} \neq M_2'''|_{P_2}, \quad (33d)$$

$$s_1''', s_2''' \in (T')^*, \quad (33e)$$

$$s_3''' \in (T' \cup T_\phi^1)^*, \quad (33f)$$

where we omit a similar proof for the equivalence compared to the previous claim. Among (33b)-(33f), only (33d) is not a predicate. Using a similar construction to the one that is used to reduce the satisfiability of (29) for $\text{CC}_N(G)$ to the satisfiability of a Yen's path formula for $\text{CC}_N(G)'$, we can reduce the satisfiability of (33) to the satisfiability of a Yen's path formula for a new Petri net. Hence, the satisfiability of Item (2) for G is decidable.

(2) Proof of the hardness result:

To prove conclusion 2 of Theorem 4, we are given a Petri net $G = (N = (P, T, Pre, Post), M_0)$ and a destination marking $M \in \mathbb{N}^P$, and construct a labeled P/T net

$$G' = (N' = (P', T', Pre', Post'), M_0', T \cup \{\sigma_G\}, \ell) \quad (34)$$

as follows (see Fig. 13 as a sketch):

1. Add three places p_0, p_1, p_2 , where initially p_0 contains exactly one token, but p_1 and p_2 contain no token;
2. add three transitions t_0, t_1, t_2 , and arcs $p_0 \rightarrow t_0 \rightarrow p_0$, $t_1 \rightarrow p_1$, $t_2 \rightarrow p_2$, all with weight 1; for every $p \in P$, add arcs $p \rightarrow t_1$ and $p \rightarrow t_2$, both with weight $M(p)$;
3. add label $\sigma_G \notin T \cup \{t_0, t_1, t_2\}$, $\ell(t) = t$ for each $t \in T \cup \{t_0\}$, $\ell(t) = \sigma_G$ for each $t \in \{t_1, t_2\}$.

It is clear that if M is not covered by G then G' shown in (34) is eventually strongly detectable. If M is covered by G , then there exists a firing sequence $M_0[\sigma_1]M_1$ with $M_1 \geq M$. Furthermore, there exist two infinite firing sequences

$$\begin{aligned} &M_0'[\sigma_1]M_1'[t_1]M_2'[t_0]M_2'[t_0] \cdots , \\ &M_0'[\sigma_1]M_1'[t_2]M_2''[t_0]M_2''[t_0] \cdots , \end{aligned}$$

where $M_2' \neq M_2''$ since $M_2'(p_1) > 0$, $M_2'(p_2) = 0$, $M_2''(p_2) > 0$, $M_2''(p_1) = 0$. Also by $\ell(t_1) = \ell(t_2)$, we have G' is not eventually strongly detectable. This reduction runs in time linear of the number of places of G and the number of tokens of the destination marking M . Since the coverability problem is EXPSpace-hard in the number of transitions of G , deciding non-eventual

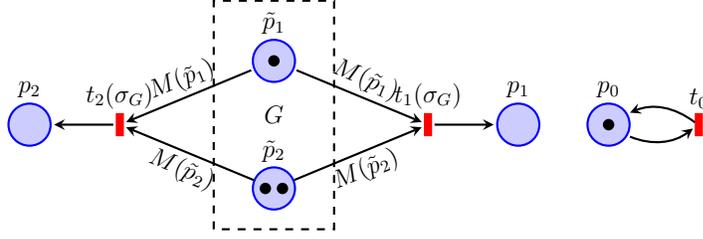


Figure 13: Sketch for the reduction in the hardness proof of Theorem 4.

strong detectability is EXPSPACE-hard in the numbers of places and transitions of G' and the number of tokens of M , hence deciding eventual strong detectability is also EXPSPACE-hard, which completes the proof. □

Remark 4 By using the extended concurrent composition and a similar procedure as the proof of Theorem 4, the decidability result for strong detectability of labeled Petri nets proved in [17] can be strengthened to hold only based on the promptness assumption.

Example 6 Consider a labeled Petri net G shown in Fig. 14, where event a can be directly observed, but b and c share the same label b . One directly sees that $\mathcal{L}^\omega(G) = \{(ab)^\omega\}$, and $\mathcal{M}(G, (ab)^n) = \{(1, 0), (0, 0)\}$ for all $n \in \mathbb{Z}_+$. Hence G is not weakly detectable, and hence not eventually strongly detectable. By its reachability graph shown in Fig. 15, one sees that this net satisfies Item (2) in the proof of Theorem 4, but not Item (1) in the proof. However, the net in Fig. 9 satisfies Item (1) but not Item (2).

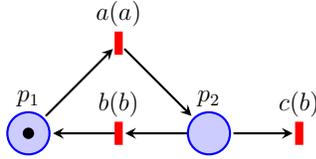


Figure 14: A labeled P/T net G .

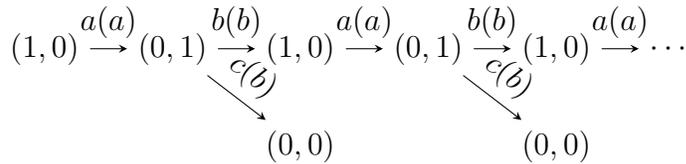


Figure 15: Reachability graph of the labeled Petri net in Fig 14.

Example 7 Consider a labeled Petri net G shown in Fig. 16. Its reachability graph is shown in Fig. 17, one has $\mathcal{L}^\omega(G) = \{ab^\omega\}$. By the reachability graph, one sees that the net is not prompt, since there is a repetitive firing sequence in $(1, 0, 0, 0, 0) \xrightarrow{t_2(a)} (0, 0, 1, 0, 0) \xrightarrow{t_4(\epsilon)} (0, 0, 1, 1, 0)$ labeled by the empty string. This net is not eventually strongly detectable, since for each $n \in \mathbb{Z}_+$, $|\mathcal{M}(G, ab^n)| = \infty > 1$. However, the net does not satisfy Item (1) or Item (2) in the proof of Theorem 4.

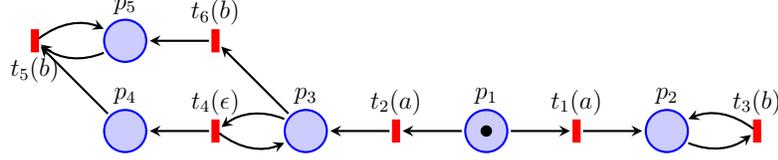


Figure 16: A labeled P/T net G .

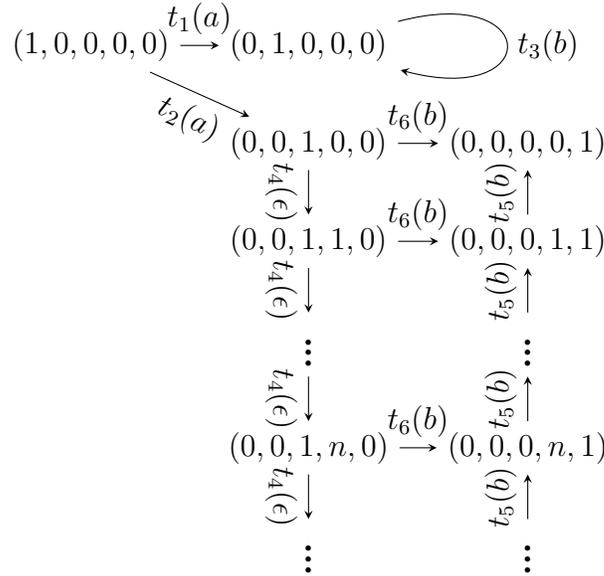


Figure 17: Reachability graph of the labeled Petri net in Fig 16.

5 Conclusion

In this paper, we obtained a series of results on detectability of discrete-event systems. We proposed one new notion of weak detectability and one new notion of strong detectability. We proved that (1) the problem of verifying weak approximate detectability of labeled Petri nets is undecidable; (2) the problem of verifying eventual strong detectability of labeled Petri nets is decidable and EXPSPACE-hard under the promptness assumption; (3) for finite automata, the problem of

verifying weak approximate detectability is PSPACE-complete, and the other property can be verified in polynomial time. (4) The relationships between these notions of detectability were also characterized, and it was proved that no two of them are equivalent.

The decidability of strong detectability and eventual strong detectability of labeled Petri nets without the promptness assumption are two interesting open problems. It is also an open problem whether there is a reduction from weak detectability to weak approximate detectability, or vice versa, for labeled Petri nets. Other variants of notions of detectability, e.g., different notions of approximate detectability are left for further study. Uniform versions of these notions of detectability are left for further study. It is also an interesting topic to look for fast algorithms for verifying these notions for (bounded) labeled Petri nets.

References

- [1] R. Ammour, E. Leclercq, E. Sanlaville, and D. Lefebvre. Faults prognosis using partially observed stochastic Petri-nets: an incremental approach. *Discrete Event Dynamic Systems*, 28(2):247–267, Jun 2018.
- [2] M. F. Atig and P. Habermehl. On Yen’s path logic for Petri nets. In Olivier Bournez and Igor Potapov, editors, *Reachability Problems*, pages 51–63, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.
- [3] 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.
- [4] K. Cai and W. M. Wonham. *Supervisor Localization: A Top-Down Approach to Distributed Control of Discrete-Event Systems*. Springer International Publishing, 2016.
- [5] L. E. Dickson. Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors. *American Journal of Mathematics*, 35(4):413–422, 1913.
- [6] D. F. Escrig and C. Johnen. Decidability of home space property. *Rapport no. LRI-503, Laboratoire de Recherche en Informatique, University of South Paris, Orsay*, 07:1–14, 1989.
- [7] E. Fabre, L. H elou et, E. Lefauchaux, and H. Marchand. Diagnosability of repairable faults. *Discrete Event Dynamic Systems*, 28(2):183–213, Jun 2018.
- [8] E. Fornasini and M. E. Valcher. Observability, reconstructibility and state observers of Boolean control networks. *IEEE Transactions on Automatic Control*, 58(6):1390–1401, June 2013.
- [9] A. Giua and C. Seatzu. Observability of place/transition nets. *IEEE Transactions on Automatic Control*, 47(9):1424–1437, Sep 2002.
- [10] M. Hack. Petri net languages. Technical report, Cambridge, MA, USA, 1976.

- [11] P. Jančar. Decidability questions for bisimilarity of Petri nets and some related problems. In Patrice Enjalbert, Ernst W. Mayr, and Klaus W. Wagner, editors, *STACS 94*, pages 581–592, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg.
- [12] C. Keroglou and C. N. Hadjicostis. Detectability in stochastic discrete event systems. *Systems & Control Letters*, 84:21–26, 2015.
- [13] S. R. Kosaraju. Decidability of reachability in vector addition systems (preliminary version). In *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing*, STOC '82, pages 267–281, New York, NY, USA, 1982. ACM.
- [14] J. L. Lambert. A structure to decide reachability in Petri nets. *Theoretical Computer Science*, 99(1):79–104, 1992.
- [15] R. J. Lipton. *The reachability problem requires exponential space*. Research report. Department of Computer Science, Yale University, Yale University. Department of Computer Science edition, 1976.
- [16] T. Masopust. Complexity of deciding detectability in discrete event systems. *Automatica*, 93:257–261, 2018.
- [17] T. Masopust and X. Yin. Deciding detectability for labeled Petri nets. *Automatica*, 104:238–241, 2019.
- [18] E. W. Mayr. An algorithm for the general Petri net reachability problem. *SIAM Journal on Computing*, 13(3):441–460, 1984.
- [19] C. E. V. Nunes, M. V. Moreira, M. V. S. Alves, L. K. Carvalho, and J. C. Basilio. Codiagnosability of networked discrete event systems subject to communication delays and intermittent loss of observation. *Discrete Event Dynamic Systems*, 28(2):215–246, 2018.
- [20] J. E. Pin and D. Perrin. *Infinite Words: Automata, Semigroups, Logic and Games*. Elsevier, 2004.
- [21] C. Rackoff. The covering and boundedness problems for vector addition systems. *Theoretical Computer Science*, 6(2):223–231, 1978.
- [22] C. Reutenauer. *The Mathematics of Petri Nets*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1990.
- [23] Y. Ru and C. N. Hadjicostis. Sensor selection for structural observability in discrete event systems modeled by Petri nets. *IEEE Transactions on Automatic Control*, 55(8):1751–1764, Aug 2010.
- [24] 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.

- [25] Y. Sasi and F. Lin. Detectability of networked discrete event systems. *Discrete Event Dynamic Systems*, 28(3):449–470, Sep 2018.
- [26] D. Sears and K. Rudie. On computing indistinguishable states of nondeterministic finite automata with partially observable transitions. In *53rd IEEE Conference on Decision and Control*, pages 6731–6736, Dec 2014.
- [27] S. Shu and F. Lin. Generalized detectability for discrete event systems. *Systems & Control Letters*, 60(5):310–317, 2011.
- [28] S. Shu and F. Lin. Delayed detectability of discrete event systems. *IEEE Transactions on Automatic Control*, 58(4):862–875, April 2013.
- [29] S. Shu and F. Lin. I-detectability of discrete-event systems. *IEEE Transactions on Automation Science and Engineering*, 10(1):187–196, Jan 2013.
- [30] S. Shu, F. Lin, and H. Ying. Detectability of discrete event systems. *IEEE Transactions on Automatic Control*, 52(12):2356–2359, Dec 2007.
- [31] M. Sipser. *Introduction to the Theory of Computation*. International Thomson Publishing, 1st edition, 1996.
- [32] R. Su and W. M. Wonham. Supervisor reduction for discrete-event systems. *Discrete Event Dynamic Systems*, 14(1):31–53, Jan 2004.
- [33] S. Takai and R. Kumar. Implementation of inference-based diagnosis: computing delay bound and ambiguity levels. *Discrete Event Dynamic Systems*, 28(2):315–348, Jun 2018.
- [34] A. F. Vaz and W. M. Wonham. On supervisor reduction in discrete-event systems. *International Journal of Control*, 44(2):475–491, 1986.
- [35] W. Wang, S. Lafortune, and F. Lin. An algorithm for calculating indistinguishable states and clusters in finite-state automata with partially observable transitions. *Systems & Control Letters*, 56(9):656–661, 2007.
- [36] X. Xu and Y. Hong. Observability analysis and observer design for finite automata via matrix approach. *IET Control Theory Applications*, 7(12):1609–1615, Aug 2013.
- [37] H. C. Yen. A unified approach for deciding the existence of certain Petri net paths. *Information and Computation*, 96(1):119–137, 1992.
- [38] X. Yin. Initial-state detectability of stochastic discrete-event systems with probabilistic sensor failures. *Automatica*, 80:127–134, 2017.
- [39] X. Yin. Verification of prognosability for labeled Petri nets. *IEEE Transactions on Automatic Control*, 63(6):1828–1834, June 2018.
- [40] X. Yin and S. Lafortune. On the decidability and complexity of diagnosability for labeled Petri nets. *IEEE Transactions on Automatic Control*, 62(11):5931–5938, Nov 2017.

- [41] X. Yin and S. Lafortune. Verification complexity of a class of observational properties for modular discrete events systems. *Automatica*, 83:199–205, 2017.
- [42] X. Yin and S. Lafortune. Minimization of sensor activation in decentralized discrete-event systems. *IEEE Transactions on Automatic Control*, 63(11):3705–3718, Nov 2018.
- [43] T. Yoo and H. E. Garcia. Event diagnosis of discrete-event systems with uniformly and nonuniformly bounded diagnosis delays. In *Proceedings of the 2004 American Control Conference*, volume 6, pages 5102–5107, June 2004.
- [44] K. Zhang. The problem of determining the weak (periodic) detectability of discrete event systems is PSPACE-complete. *Automatica*, 81:217–220, 2017.
- [45] K. Zhang and A. Giua. Weak (approximate) detectability of labeled Petri net systems with inhibitor arcs. *IFAC-PapersOnLine*, 51(7):167–171, 2018. 14th IFAC Workshop on Discrete Event Systems WODES 2018.
- [46] K. Zhang, T. Liu, and D. Cheng. Observability of finite labeled transition systems. *IEEE Transactions on Automatic Control*, 63(6):1501–1512, June 2018.
- [47] K. Zhang and L. Zhang. Observability of Boolean control networks: A unified approach based on finite automata. *IEEE Transactions on Automatic Control*, 61(9):2733–2738, Sept 2016.
- [48] K. Zhang, L. Zhang, and R. Su. A weighted pair graph representation for reconstructibility of Boolean control networks. *SIAM Journal on Control and Optimization*, 54(6):3040–3060, 2016.